

# 14

## NOISE AND SONIC BOOM IMPACT TECHNOLOGY

Initial Development of an Assessment System for Aircraft Noise (ASAN): Software Listing

Volume IV of IV Volumes

Sanford Fidell Nicolaas Reddingius Michael Harris B. Andrew Kugler

BBN Systems & Technologies Corporation 21120 Vanowen Street Canoga Park, CA 91303

June 1989

Final Report for Period February 1987 - October 1988

Approved for public release; distribution is unlimited.

Noise and Sonic Boom Impact Technology Program
Human Systems Division
Air Force Systems Command
Brooks Air Force Base, TX 78235-5000

19980925 00(

#### NOTICES

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility nor any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise as in any manner construed, as licensing the holder, or any other person or corporation; or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

The Office of Public Affairs has reviewed this report and it is releasable to the National Technical Information Service (NTIS), where it will be available to the general public, including foreign nationals.

This report has been reviewed and is approved for publication.

ROBERT C. KULL, JR, Capt, USAF NSBIT Program Manager

FOR THE COMMANDER

MICHAEL G. MACNAUGHTON, COL, USAF

Deputy Commander Development & Acquisition

Please do not request copies of this report from the Human Systems Division. Copies may be obtained from DTIC. Address your request for additional copies to:

Defense Technical Information Center Cameron Station Alexandria VA 22301-6145

If your address has changed, if you wish to be removed from our mailing list, or if your organization no longer employs the addressee, please notify HSD/SORT, Brooks AFB TX 78235-5000, to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE							Form Approved OMB No. 0704-0188	
1a. REPORT SECURITY CLASSIFICATION Unclassified				1b. RESTRICTIVE MARKINGS				
2a. SECURITY CLASSIFICATION AUTHORITY				3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution				
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE				is unlimited				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Project 04515, Report 6624, NSBIT Task Order				5. MONITORING ORGANIZATION REPORT NUMBER(S) HSD-TR-89-010 Vol. IV				
6a. NAME OF PERFORMING ORGANIZATION BBN Systems & Technologies  6b. OFFICE SYMBOL (If applicable)				7a. NAME OF MONITORING ORGANIZATION				
Corr	coration	-	( 0,)	Advanced Development Program Office HSD/YA-NSBIT				
6c. ADDRESS (City, State, and ZIP Code) 21120 Vanowen Street Canoga Park, CA 91303				7b. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB OH 45433-6573				
ORGANIZATION Advanced Develop-				9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER				
ment Program Office HSD/YA-NSBIT 8c. ADDRESS (City, State, and ZIP Code)				F33615-86-C-0530  10. SOURCE OF FUNDING NUMBERS				
	atterson A			PROGRAM	PROJECT	TASK WORK UNIT		
OH 45433-6573.				ELEMENT NO.	NO.	NO.	ACCESSION NO.	
11 TITLE (Incl	lude Security C	(lassification)		63723F	3037	02	01	
11. TITLE (Include Security Classification) Initial Development of an Assessment System for Aircraft Noise (ASAN), Volume IV: Source Code Listings								
12 PERSONAL AUTHOR(S) Fidell, Sanford; Reddingius, Nicolaas; Harris, Michael, and Kugler, B. Andrew								
13a. TYPE OF Final		13b. TIME CO		4. DATE OF REPORT (Year, Month, Day) 15. PAGE COUNT 276				
16. SUPPLEMENTARY NOTATION								
17.	COSATI		18. SUBJECT TERMS (C	Continue on reverse if necessary and identify by block number)				
FIELD	GROUP	SUB-GROUP	source code co	mouter progr	am ASAN			
12	07			mpuoer program ranti				
19. ABSTRACT (Continue on reverse if necessary and identify by block number)  This is the fourth volume of a volume report summarizing the development and current contrents of a preliminary prototype version of an Assessment System for Aircraft Noise (ASAN). ASAN is a computer-based system intended to assist members of the United States Air Force (USAF) environmental planning community in addressing noise-related issues in developing environmental impact analysis documents, in compliance with USAF and other regulations, especially the National Environmental Policy Act (NEPA) of 1969.  This volume contains technical appendices and listings of the source code for the preliminary prototype version of ASAN.								
I		LITY OF ABSTRACT		1. ABSTRACT SECURITY CLASSIFICATION				
	SIFIED/UNLIMIT		PT. DTIC USERS		Unclassified  2b. TELEPHONE (Include Area Code)   22c. OFFICE SYMBOL			
22a NAME OF RESPONSIBLE INDIVIDUAL Robert C. Kull, Captain USAF				(513) 255-338			YA-NSBIT	

# **Table of Contents**

APPENDIX A. SCREEN DESCRIPTION FILES	1
<ul> <li>A.1 Screen Description File for Introductory Portions of ASAN</li> <li>A.2 Screen Description File for MTR-related Operations</li> <li>A.3 Screen Description File for Report-Related Portions of ASAN</li> <li>A.4 Screen Description File for Graphic Portion of ASAN</li> </ul>	1 24 102 106
APPENDIX B. PROGRAM LISTINGS	117
B.1 C Language Source Code	252

# Appendix A SCREEN DESCRIPTION FILES

This appendix contains the screen description files from which the U parser produces C code that can be compiled and linked with other object modules to produce an executable image. These files describe the appearance of the user interaction screens, and also determine what action the program takes when users elect specific actions.

### A.1 Screen Description File for Introductory Portions of ASAN

```
MY FOR THE ASAM typedef DEFINITIONS MEET
        COMPILE USUSS.C WITHOUT INCORDING THE WRATE OF THE COMPILER
INCLUDE ASSETTIVE .
 DECLARATIONS FOR INTRODUCTIONY SCHOOL
 PERFELOCE introtet2 (
   ENE_POWS 8
SWE_COLUMNS 76
                 tstblk/istrotst.tst
 TEMPOW introvindow1 (
        columns 78
    textblock introtat2, 1, 1
VARIABLE enimals (
           DATOM enimels (
      EWE TOWN
      verisble
      pickeble
                  100
      trailer
                   "enimals remain in list"
VARIABLE qual_estries (
type INTEGER
formet 45d
   DATCH qualif (
      aum_columns 32
                 qual_entries, 0, 10
      pickable
                  "Thus far"
      trailer
                  "entries qualify"
   DATUM aumofeit (
     ME TOWN
      Pariable qual entries, 0 , 0 trailer citations meeting
      trailer
```

```
pickable MO
  VARIABLE GEFFGIT (
     type INTEGER
     DATOM ourreit (
        REEL FORM 1
         variable curreit, 0 , 22
leader "Display shows number"
        leeder "Disp
trailer "of"
pickable HO
  VARIABLE when {
foundis "VARCEAR datep.arr"
     type STRING
     formet 14s
     DATOM when (
        nun rows 1
nun columns 16
        variable when, 0 , 11
leader "Published"
        Leader
        pickable 20
  TEXTLINE cancechtst ("CANCEL SEARCE")
           cancech (
     am_rows 1
     aum_columns 17
    textline cancechtzt, 0,2
helpfile help/cmclsrch.hlp
 TEXTLINE contactst ("Interrogate point-of-contact database")
 BUTTON contacts {
    aum_columns 40
textline contactxt, 0 ,2
helpfile help/contacts.hip
 TEXTLINE humantit ("Interrogate human effects citation index")
 BUTTON human (
     ava_columns 50
    tertline humantzt, 0 ,2
helpfile help/humcit.hlp
TEXTLES unimeltat ("Interrogate animal effects ditation index")
BUTTOW aminal (
    See Lone
     aum_columns 50
    textline animaltat, 0 ,2
helpfile help/assist.hip
TEXTLES structut ("Interrogate structural effects citation index")
BOTTOM stract (
   NUM_TOWN 1
    textline structut, 0 ,2
helpfile help/stroteit.hlp
TEXTLES somistit ("Interrogate noise & somis boom modeling sitution index")
MUTTON sonic (
    num_columns 60
    textline
                somictat, 0 ,2
```

```
helpfile help/somic.hlp
 THEFAIRE legislatut ("Interrogate legislative database")
 BUTTOW legislat (
     am_ross 1
      num_columns 40
    textline legislatzt, 0 ,2
helpfile help/legislat.hlp
 TEXTAINS selectedint {
"You can now interrogate these ASAN detabases:"}
 WINDOW inquiry (
     SEE TOWN
      awn_oolwas
                     78
     محلا
                     0,0, 0,77
     textline
                                    3,10, "CALL quetup",
"NEW SCREEN contactscreen"
     button
                     contacts,
     button
                     henes,
                                    4,10, "CALL quetup",
                                           "MEW SCHOOL heitlesen"
     button
                                    5,10, "CALL quetup",
                     enimal.
                                           "MEN SCREEN soitlegra"
                                    6,10, "CALL quetup",
"NEW_SCREEN scitisors"
     button
                     struct,
     button
                                    7,10, "CALL quetup",
                     sonis,
                                           "MEN SCREEN moitleors"
     button
                    legislat,
                                    0,10, "HEN_SCREEN Legislatscreen"
 ) paids waste
    mainscreen THS
    border
                 THE
    window
                 inquiry,
                              10,1
     window
                 majoraction4, 19,1
introvindow1, 1, 1
                   Declarations for POINT OF CONTACT DATABASE SCREEN
TENTLINE streezchtzt ("START SEARCE")
BUTTON streams (
   Name 1
   hum_columns 15
   textline stsearchtxt, 0 ,2
helpfile help/stsearch.hlp
TENTINE caff016tst ("AFT-014 attributes")
BUTTON caff814 (
   RUM_rows 1
   RUM COLUMN 20
   textline ceff814txt, 0 ,2
helpfile help/ceff814.hlp
TEXTLES otribultat ("Tribul")
BUTTOW Stribel {
   num_rows 1
   num columns 9
   textline otribultat, 0 ,2
helpfile help/otribul.hlp
TENTEDER omilitarytet ("Military")
MOTTOW cailitary (
   num_rows 1
   aum_columns 11
  textline cuiliterytet, 0 ,2
helpfile help/cuiliter.hlp
```

```
TEXTLIRE ofederaltst ("Federal")
   BOTTOM ofederal (
      sum_rows 1
      aum_columns 10
textline ofederaltxt, 0 ,2
helpfile help/ofederal.hlp
  TEXTLINE ostatetat ("State")
  SUFFICE estate (
     aum_rows 1
aum_columns $
     textline estatetxi, 0 ,2
helpfile help/ostate.hlp
  TEXTLES cocurtytat ("County")
  BUTTON cocusty (
     aum rome 1
      awa_columns $
     tartline cocuntytzt, 0 ,2
helpfile help/cocunty.hip
  TEXTLINE ocitytat ("City")
  SUTTON GGILY (
     num rows 1
     textline ocitytet, 0 ,2
helpfile help/coity.hlp
 VARIABLE address (
   type STRING
format 4-25s
 DATCM address {
    aum_rows 1
    aum_oolumns 77
       variable address, 0 ,28
leader "City and/or state address: "
helpfile help/address.hlp
 VARIABLE contain (
type STRING
format 4-15s
DARTM contnam {
   aum_rows 1
   aum_columns 77
      variable contamn, 0 ,28
leader "Point of contact last name:"
helpfile help/contamn.hlp
VARIABLE affealog (
    type STRING
format 4-15s
    DATUM affected {
        aum_rows 1
        aven_columns 50
       variable effseled, 0 , 33
leader "Affiliation currently selected:"
TEXTLINE agentst ("Agency:")
TEXTLINE searchtst ("Search by:")
WINDOW contactsearch {
       num_rows 17
num_columns 78
```

```
line
                          0,0, 0,77
                           searchtst, 1, 1
          textline
           detun
                           ocetzen,
                                         3, 1
                           address,
           detun
                                         4, 1
          textline
                          egentat,
                                         5, 1
          button
                           coity,
                                                "CALL setaff 0"
                                         B, 10,
          buttos
                          ecounty,
                                         5,10, "CALL setaff 1"
          button
                           estate,
                                         5,27, "CALL setaff 2"
          button
                          ofederal,
                                         5,37, "CALL setaff 3"
          buttoe
                          omilitary, 5,40, "CALL setaff 4"
          button
                          otribel,
                                         8, 60,
                                                 "CALL setaff 5"
          deten
                          affseled, 6, 1
caffs14, 9, 1, "MEN SCHEM contstsearch"
stsearch, 16, 1, "CALL possch"
ceanosch, 16,45, "CALL quetup",
                          affseles,
          button
          button
          button
                                                  THE SCHOOL SPINS
         }
 SCHEET contactscreen { title "POINT OF CONTACT DATABASE"
                    contactsearch, 2,1
majoractions, 19,1
      window
            Point of contact attribute search screen
 TRATLINE minoretzt ("Minor ettributes")
 TEXTLIES scrolltxt ("Scroll window for more alternatives")
  TENTRICCE minorat (
      filename tutblk/minor.tut
mum_rows 10
mum_columns 35
       border
                         YES
 WINDOW minorat (
      textline minoretx, 1,0
textline minoretx, 2,0
textline scrolltxt, 12,0
THEFFICIE majoret (
filename txtblk/major.txt
hum_rows 10
hum_columns 35
VARIABLE minorat (
   type STRING
format 4-45s
DASTM minoret (
    AUM_ROWS 1
      aum_columns
     variable minoret, 0 ,22
leader "Type minor attribute:"
helpfile help/minor.hlp
VARIABLE majoret (
     type STRING
formet 4-15s
DATOM majorat (
     aum rows 1
     num columns
     variable majoret, 0 ,22
leader "Type major attribute:"
     leader "Type major att
helpfile help/major.hlp
```

```
)
   TEXTLEM majoratet ("Major attributes")
            contatemental (
         BUR TOWN
                      17
78
          aum_columns
          محلا
                       0,0, 0,77
                                    1, 1 -, "ADD_WIMDOW minorat 8 41"
2, 1 -, "GALL demay"
          datum
                       majorat,
          dete
                       minorst,
          textline
                       majoretzt,
                                   4, 1
          textline
                       scrolltst, 15, 1
          textblock
                       majoret,
                                    8. 1
          testhlosk
                       minorat,
                                    8,41
         button
                       steesroh,
                                   16, 1,- "REMOVE WINDOW"
                                   "CALL postob"
16,45, "CALL quetup",
         button
                       canosch,
                                          THEN SCHOOL COLLEGE
        )
  SCREEN contatsearch (
      title
                  "POINT OF CONTACT ATTRIBUTE SHANCE"
               contatsearch, 2,1
majoraction4, 19,1
YES
      Window
      Window
     border
           Point of contact display screen
  wassaugur: all of these pos variables need to be declared in a header file
 VARIABLE F_MANG {
foundia "VARCHAR f_name.arr"
                     STRUM
    type
                       4-10s
 DATOM contactness: {
    am_rows
                       25
                       P HOME, 0, 15 "Contact Home:"
    variable
    leader
    pickeble
 VARIABLE L MANG {
foundin "VARCHAR L name.arr"
    type
    formet
                       1-15€
)
DATOM contactness2 (
   atte columns
   veriable
                      L_MME, 0, 1
   pickable
VARIABLE CONTITUE ( -TITLE may be a reserved word
   foundia "VARCHAR contitle.err"
                  STRING
   type
                      4-45s
DATOM contitle (
   FART TORN
   awa columns
                      70
   variable
                      CONTINUE, 0, 9
   leeder
                      "Title:"
  pickable
VARIABLE OFFICE {
  foundin "VARCEAR office.arr"
  type
                      977LU93
```

```
4-45e
  DATOM office {
     RIM_POWS
                          70
     veriable
                          OFFICE, 0, 15
     leader
                          "Office: "
     pickable
 VARIABLE AGENCT DEPT {
foundia "VARCHAR agency_dept.arr"
symples
                          4-45s
  DATOM agency (
    num columns
                         1
                         70
     verieble
                         ACTION DEST. 0, 10
                          "Agency: "
    pickeble
 VARIABLE ST AND DIV { "***cally 25 characters for a street address? foundia "VARCHAR st_add_div.arr" type STRING
    type
format
                         4-25a
 )
 DATCH st_add_div {
    am column
                         50
     veriable
                        ST ADD DIV, 0, 17
    leader
                         "Street Address:"
   pickable
 VARIABLE PO_BOX (
    foundin "VARCHAR po box.arr"
    type
                         FTRUM
    formet
                         4-10s
 DATOM po_box {
    aw_column
                        40
    Variable
                        PO_BOX, 0, 18
    leader
                        "Post Office Box:"
   picksble
                        200
 VARIABLE MISC ADD {
    foundin "VARCEAR misc_add.arr"
    type
                       FIRING
    format
                        4-10s
DATOM misc_add (
   RUM_FOUR
                       20
   variable MISC ADD, 0,2
   pickable
                       100
VARIABLE CITY_BASE {
foundia "VARCHAR dity_base.arr"
type #-25s
DATOM city_base (
   BEER FOWS
   Man columns
                       30
                       CITY_BASE, 0, 1
   pickeble
VARIABLE STATE (
foundin "VARCEAR state.arr"
   type
```

579.IIIG

-

```
formet
                         4-20
   }
   DAFTM state (
      2000
      2001
  piskable
piskable
      variable
                         SEATE, 0, 1
   WARIABLE SIPCODE (
     foundia "VARCEAR Eipoode.arr"
      type
                        STRING
     formet
  DATOM Expende {
    Run_columns
                        1
                        10
     variable
                        EXPOODE, 0, 1
    piskable
  VARIABLE MAIL_CODE (
     foundin "VARCEAR mail code.arr"
                     STR DIG
     type
formet
                        4-10s
  DATTM mail_code (
    BEEL FORE
     num_columns
veriable
                        40
                       MAIL CODE, 0, 12
Thail Code:
     leader
    pickable
  VARIABLE PEOME {
    foundia "VARCEAR phone.arr"
type #FRIES
  }
 DATOM phone (
    AUM_CONS
                       40
    variable
                       PECHE, 0, 20
    leader
                       "Telephone Number:"
   piakabla
 VARIABLE APPILIATIO (
    foundin "VARCEAR affiliatio.arr"
   type
formet
                    STRING
                       4-80
 DATOM affiliatio (
                     1
30
   NUM_FORM
    variable
                    AFFILIATIO, 0, 17
  pickable
VARIABLE MAJOR_ATTRIB (
   foundin "VARCEAR major_attrib.arr"
   type
                    STRING
   formet
                     4-16s
DATEM major_attribute {
  PART TOWN
   nun_columns
   Variable MAJOR ATTRIB, 0, 19
leader "Major Attribute:"
pickable 20
  leader
                     100
VARIABLE MINOR_ATTRIBUTE (
  foundin "VARCEAR minor_attribute.arr"
```

type

FRING

```
format
                          4-45e
   DATOM minor_sttribute {
      NE TOWN
     SWE_COLUMNS
Variable
                          MINOR ATTRIBUTE, 0, 19
                         "Minor Attribute:"
                         100
  VARIABLE AREA (
     foundin "VARCEAR area.arr"
     type
                        STRING
                          4-15e
  DATOM area {
     AWE TOWN
                         1
     num columns
                         30
                         AREA, 0, 7
"AREA:" -- (whatever that meens)
   " leeder
    pickable
  VARIABLE SCORE (
     foundin "VARCHAR scope.arr"
     type
                         FERRING
                         9-10s
 DATOM scope (
    NUM COLUMNS
                        1
                        30
                      SCOPE, 0, 8
"Scope:" -(whetever that means)
     variable
  ~ leader
    pickable
 WINDOW posstuff (
aum_rows 17
aum_columns 78
line 0,0
                    0, 0, 0,77
contactame1,
contactame2,
     detun
                                        1, 2
     detre
                                    1,40
2, 2
     detun
                    contitle,
     detus
detus
                    office,
                                        3. 2
                    egency,
st_edd_div,
                                        4, 2
     detun
     detun
                   po_boz,
     detus
                                        €, 25
     detw
                    city_bese,
                                       7, 2
7,25
     detus
                    state,
     detus
                    Elpoode,
                                        7,35
     detu
                   mail_dode,
                                        B. 2
     detun
                   phone,
                   affiliatio,
                   major_attribute, 10, 2
     detun
     detus
                   misor sttribute,
                                      11, 2
12, 2
     detun
                   eres,
    detun
                   scope,
                                      12,40
TENTETE nexteddr ("Show next address")
BOTTON mextaddr (
   am_rows 1
   ave columns 25
   tartline nexteddr, 0, 2
helpfile help/sohelp.hlp
WINDOW getoffscreen {
    aum rows 4
    aum columns 78
                 1120
    button
    betton
   betton
```

```
.mm scame epind.
  SCREEN poodisplay (
title "DISPLAY POINT OF CONTACT INFORMATION"
        window
                           pocetuff, 2,1
getoffscreen, 18,1
        window
        border
              Numer effects citation detabase screen
  VARIABLE titlefreg (
     foundin "VARCHAR titlefreg.arr"
type STAING
formet 4-60s
     DARTH titlefreg (
        htm_rows 1
         ave columns 65
         variable titlefreg, 0 ,0
helpfile help/titlfreg.hlp
 TEXTLEME keywrdtzt ("Eeyword categories")
 MOTTON keywid (
     EWOT_MEE
     num_columns 25
tartline keywrdtst, 0, 2
helpfile help/keyword.hlp
 VARIABLE dates2 {
    type INTEGER
formet 44d
    )
DATCH dates2 (
       FER LONE
        ava_columns 15
        variable dates2, 0 ,4
        leeder
trailer
                        "and"
                    "(Jecz) "
"Syd.
                    help/dete.hlp
        belpfile
VARIABLE dates1 {
   type INTEGER
   formet 44d
    DATEM detes1 {
       aum rows 1
aum columns 27
       variable dates; 0 ,14
leader "Date between:"
trailer "(year)"
helpfile help/date.hlp
VARIABLE authornes {
   foundin "VARCEAR authornen.arr"
   type STRING
formet 4-40s
   DATOM author (
       num columns 77
       variable authorsem, 0 ,20
Leader "Author's last name:"
       helpfile help/author.hlp
VARIABLE citattemi (
                                                            - Note: The string lengths
   foundin "VARCEARAO authorlist[0].arr"
                                                            " are to get by the parser.
" The progress will figure out " whether to allow 30 or 40
   type STRING
formet 4-30s
                                                            -columns on the display.
   DATOM citestnemi (
      num rows 1
num columns 40
```

```
mitautnemi, 0, 9
"Author:"
        veriable
        leeder
        pickable
                       20
 YARIABLE citestaem2 (
     foundia "VARCHAR40 authorlist[1].arr"
     type STRING
formet 4-30s
     DATOM citautaem2 (
        REAL POWS 1
        ava_columns 40
        variable citavtsem2, 0, 9
pickeble 20
        pickeble
 VARIABLE ditestment (
    foundin "VARCEAR40 authorlist[2].arr"
    type STRING
format 9-30s
    DATOM citestanni (
       aum rows 1
num columns 40
       variable attautama, 0, 9
pickable 20
VARIABLE citestaen4 {
    foundia "VARCEAR40 authorlist[3].arr"
    type STRING
format 4-30s
    DATOM citastami (
       ama_rows 1
ama_columns 40
variable citautamn4, 0, 9
pickable 20
VARIABLE citestaen5 (
    foundia "VARCEAR40 authorlist[4].arr"
    type STRING
formet 4-30s
    DATOM citewtness (
       aum_rows 1
aum_columns 40
       variable ditautness, 0, 9
pickable 20
VARIABLE citautame (
   foundin "VARCHARAO authorlist[5].arr"
type STRING
formet 4-20s
   DATOM citestness {
      RUM FORE 1
aum_columns 23
variable citautamns, 0, 0
pickable 20
VARIABLE citestaem? {
   foundin "VARCHAR40 authorlist[6].arr"
   type STRING
format 4-20s
   DATOM citestaum? (
      aum columns 23
      variable diteutaem7, 0, 0 pidmble MO
VARIABLE citautamns (
   foundin "VARCEAR40 authorlist[7].arr"
  type STRING
format 4-20s
  DATCM citautamns (
```

```
num_rows 1
num_columns 23
         variable citattems, 0, 0
         pickeble
  VARIABLE Gitautames (
     foundin "VARCHAR40 authorlist[0].arr"
     type STRING
format 4-20s
     DATOM citestams {
        num_rows 1
num_columns 23
        variable ditautaems, 0, 0 pickable 20
 VARIABLE ditentmenio (
     foundin "VARCHAR40 authorlist[9].arr"
     type STRING
format 8-20s
     DATOM citertannio (
        num_rows 1
num_columns 23
        variable citautnam10, 0, 0
pickable 20
 TEXTLES estitlist ("Title:")
 WINDOW heitlearn (
    aum_rows 16
aum_columns 78
    1120
                    0, 0, 0,77
    textline
                  searchtst, 1, 1
                   qualif, 1,45
    detus
                   author,
                                 3, 1, "CALL VCAPITAL Sauthornam",
                                         "MENVALE"
                   entititrt, 4, 1
titlefreg, 4, 9, "CRIL VCAPITAL Stitlefreg",
    textline
    detun
                  dates1, 8, 1
dates2, 8,27
keywrd, 10, 1, "CALL herch010"
stsearch, 15, 1, "CALL herch001"
oandsch, 15,45, "CALL quetup",
"ERW_SCREEN dbing"
                                         "MENVALE"
    detum
    button
    bettos
    button
   }
SCHOOL Moitisons (
   title "SUMAN EFFECTS CITATION SHARCE"
   window
                hoitlearn, 2,1
  window
                 majoractica3, 18,1
              724
   border
- Declarations for animal effects citation detabase screen
WINDOW acitisora (
   RUM_rows 16
RUM_columns 78
                  0,0, 0,77
   textline
                  searchtzt, 1, 1
   detu
                  qualif, 1,45 author, 3, 1, "CALL VCAPITAL Sauthornem",
   deten
                                        "MEWVALE"
                  estitliri, 4, 1
titlefrag, 4, 9, "CALL VCAPITAL Stitlefrag",
   textline
                                        "MEWVALE"
                             0, 1
8,27
   deton
                  detes1,
                  dates2,
   detus
                 datas2, 8,27
ksywrd, 10, 1, "CALL asrch010"
stsearch, 15, 1, "CALL asrch001"
cancsch, 15,45, "CALL quetup",
"EER_SCHOOL dbing"
   button
   button
   button
```

```
SCREEN aciticora (
       title
                  "ANTHOL EFFECTS CITATION SEARCE"
       Window
                  acitiscra, 2,1
        Window
                  majorestical, 18,1
          Declaration for structural effects citation database screen
 WINDOW saitisara
                   16
      awa_columns 76
                   0,0, 0,77
      1420
      textline
                    searchtst, 1, 1
      detu
                    qualif,
                               1,45
       detun
                    author,
                             3, 1, "CALL VCAPITAL Seuthornen",
                                      "MENVALS"
      textline
                    emtititut, 4, 1
                   titlefreg, 4, 9, "CALL VCAPITAL Stitlefreg",
                                      "MANVALE"
      detu
                             8, 1
      detre
                    dates2,
                               8,27
                   keywrd, 10, 1, -*CALL serob010*
                                      CYLL quest.
                   stsearch, 15, 1, "CALL serch001" cancech, 15,45, "CALL quetup", "MEN_SCREEN chinq"
      button
 SCHOOL SCITISON (
      title
              "STRUCTURAL EFFECTS CITATION SEARCH" -ar 2/4 Corrected
                 scitiscra, 2,1
majoraction3, 18,1
      window
       window
      border

    Declaration for noise and scale boom modeling effects citation database

 TEXTLINE motimplement ("NOT IMPLEMENTED IN PROTOTYPE VERSION OF ASAN")
WHEDOW moitison (
   BWE_TOWS
   aum columns 78
   معنة
                0, 0, 0, 77
searchtst,
    textline
                                1, 1
                 qualif,
                                1.45
                 author,
   detun
                                3, 1, "CALL VCAPITAL Sauthorness",
   textline
                 entitltmt,
                                4, 1
   datum
                titlefrag,
                                4, 9, "CALL VCAPITAL Stitlefreg",
                                      "MENVALS"
   det un
                dates1,
                                0, 1
   detun
                 dates2,
                                0,27
   button
                keywid,
                               10, 1, -*CALL merch010*
                                      "CALL dumy"
   bettoe
                              15, 1, "CALL merch001"
15,45, "CALL quetup",
"NEW SCREEN dbing"
                stseerch,
   bettoe
                onnosch,
   }
school maitison (
     title
                "MOISE AND SONIC BOOM MODELING EFFECTS SHARCE"
               maitisora, 2,1
majorastion3, 18,1
     window
      window
              120
    border
         " Declaration for LEGISLATIVE DATABASE SCREEN
WINDOW legislatecreen (
    MOZ ME
                16
```

```
202_001222 78
line 0,0,
                       0,0, 0,77
                       secretat, 1, 1
         tertline
                                  1,30
         deten
                       qualif,
                                     3, 1, "CALL VCAPITAL Sauthornes",
         detm
                       author,
                                             "MENVALS"
                       estitlist, 4, 1
titlefreg, 4, 9, "CALL WCAPITAL Stitlefreg",
        textline
         detun
                                             "MENVALS"
        dete
                                  8, 1
                       dates2,
        detun
                                    8,27
        button
                                 10, 1, "WEN SCREEN lksywrdschaureen"
                       beywrd,
                                            CALL CHARY
                       notimplement, 5,15
stsearch, 15,1, "CALL dumny"
cancech, 15,45, "CALL quetup",
         textline
         button
         bettoe
                                           THE SCHOOL SPING
        }
  SCHEM legislatscreen (
        title
                    "LEGISLATIVE CITATION DATABASE SEARCE"
        window legislatscreen, 2,1
window minuscreen, 2,1
mjoractica3, 18,1
        window
        border
          Declaration for Animal effects keyword category search screen
  VARIABLE study? {
     type STATES
formet 4-20s
     DATOM study2 (
        hum rows
        BUR Column 37
        Variable study2, 0 ,14
leader "Study type 2:"
        leader
        belpfile
                       help/study2.hlp
    DATOM study2d (
        FAM LOAD
        ama columns 37
        variable study2, 0 ,14 leader "Study type 2:" pickable mo
       pickeble
 VARIABLE study1 {
    type STRING
formet 4-20s
    DATOM study1 {
       AWE_TOWN
       aum_columns 37
       variable study1, 0 ,14
       leeder
                     "Study type 1:"
       helpfile
                   help/stady1.hlp
   )
DAROM studyld {
       EMOT TOME
      variable study1, 0 ,14
leader "Study type 1:"
VARIABLE empdess {
   foundin "VARCEAR empdess.arr"
   type STRING
formet 4-20s
  DATOM expdess (
     See Lone
      atm columns 40
      variable expdess, 0 ,19
Leader "Experimental type:"
     helpfile help/methodol.hlp
  DATOM expdesed (
```

```
BUR FOWS
          awn_columns
                            40
                           empdess, 0 ,19
"Experimental type:"
          variable
         leader
         pickable
                             200
 VARIABLE moistype {
  foundim "VARCHAR moistype.arr"
     type STRING
format 4-20s
     DARTON moistype (
         ava rows 1
ava columns 37
         Verieble
                             moistype, 0 ,12 "Moise type:"
         leader
                        help/asetype.hlp
         helpfile
     )
DATOM moistyped {
        num_rows
num_columns 37
variable acistype, 0 ,12
leader "Moise type:"
         200
 VARIABLE species4 {
   foundia "VARCHAR40 species4.arr"
     type STRING
format 4-40s
    DATOM species4 (
aum_rows
aum_columns
                           65
        variable species 4, 0, 16

Leader "Species type 4:"
helpfile help/species.hlp
    DATOM species4d (
        NUM POWS
                            1
                            65
        variable species4, 0 ,16 leader "Species type 4:"
VARIABLE species3 {
   foundin "VARCHAR40 species3.arr"
    type STRING
format 0-40s
    DATUM species3 (
      hum_rows
                           65
       variable species, 0,16
leader "Species type 3:"
helpfile help/species.hlp
   )
DATCH species3d {
       NUM TOWN
                           1
                          65
       Variable species3, 0 ,16 leader "Species type 3:"
       pickeble
VARIABLE species? (
   foundin "VARCHAR40 species?.arr"
   type #73.185
format 4-40s
   DATOM species2 (
      RUM rows 1
RUM columns 65
       variable species2, 0,16
leader "Species type 2:"
helpfile help/species.hlp
       helpfile
                           help/species.hlp
  )
DATOM species2d (
      BEEL FOWS
      aum_rows 1
aum_columns 65
```

```
species2, 0 ,16
"Species type 2:"
NO
         leeder
 VARIABLE species1 {
    foundia "VARCHAR40 species1.arr"
    type STRING
    foundt 4-40s
     DATOM species1 (
        num rows 1
num columns 65
variable species1, 0,16
leader "Species type 1:"
helpfile help/species.hlp
     DATOM species1d (
         NEW LOAD
         num_columns 65
         Variable species1, 0,16
leader "Species type 1:"
pickable 20
 TEXTRLOCK enimal (
      hum rows 6
hum columns 35
filenume triblk/animel.tri
border YES
TEXTSIOCK ver_mainel (
      RUE FOWE 8
RUE SOLUMNS 45
filenes tytblk/varanial.trt
border YES
WINDOW speciatry (
aum_rows 1
aum_solumns 50
    detum species2, 0, 1, "CALL VCAPITAL &species2",
                                               "NEWVALS",
                                              "CALL queryal2"
WINDOW speciatry (
    aum rows 1
                     species3, 0, 1, "CALL VCAPITAL Especies3",
                                               "BEWVALS",
                                             "CALL queryell"
    3
WINDOW speciatry {
   num_rows 1
   num_columns 50
             species4, 0, 1, "CALL VCAPITAL Especies4",
                                               "REWVALS",
                                              "CALL queryel4"
WINDOW
           akeywrdsearch (
   Bun Long
    aum_columns 78
                     0,0, 0,77
   1100
   detum
detum
                     qualif, 1, 1
mainels, 1,40
species1, 2, 1, "CALL WCAPITAL Sepecies1",
    detus
                                              "HENVALS",
                                              "CALL queryall"
                    species2d, 3, 1
species3d, 4, 1
species4d, 5, 1
noistype, 6, 1
expdesod, 6,38
   detun
   detun
   detus
detus
```

verieble

```
detun
                  study1d,
                              7, 1
     detun
                  study2d,
                           7,38
    textblock
                  animal.
                              0,23
    textline
                  serolitzt, 16,23
                 stream, 18, 1, "CRLL asrch002"
cancech, 18, 35, "CRLL quetup",
"MRN_SCREEN dbing"
    button
    helpfile help/sobelp.hlp
 SCREEN akeyerch {
    title "ANTHAL EFFECTS RETWOOD SEARCH"
    window akeywrdeearch, 2,1
 }
WZMDOW akeyelt
                       ( - alternate used for pop-up textblock
    rows 19
aum columns 78
line
                 0,0, 0,77
    detu
                 qualif,
                             1. 1
    detun
                 entrate,
                            1.40
    detun
                 speciesld, 2, 1
    detun
                 species2d, 3, 1
    detun
                 species3d, 4, 1
    dete
                 species4d, 5, 1
                 moistype, 6, 1, "CALL VCAPITAL Smoistype",
    dete
                                    "REWVALS",
                                    "CALL dumy"
                 empdesc, 6,38, "CALL VCAPITAL Sempdesc",
    deten
                                    "CALL dumy"
    detun
                          7, 1, "CALL dumny"
7,38, "CALL dumny"
                 studyl,
                 study2,
    detus
    textblock
                 ver animal, 8,10
    textline
                 serolltst, 16,23
    button
                 stsearch, 18, 1, "CALL asrch002"
    button
                 cencech, 18,55, "CALL quetup",
                                    MEN SCHEEN OPTER.
   belpfile
              help/zohelp.hlp
 SCHOOL aboyalt (
      title
                 "ANTIGAL REFECTS RETWORD SEARCE"
      window
                 akeyalt, 2,1
       Declaration for Ruman affects keyword category search screen
TEXTSLOCE exprtype {
                10
35
     NOT TOWN
     filenme
border
                    tatblk/methodol.txt
                    TES
WINDOW exprtype
     AWA_FOWS
                   10
                 35
     testblock
                   exprtype, 0,0
                                               ٦.
TEXTRLOCK solstype (
    aum rows
aum columns
filename
                   10
                   35
                    txtblk/asetype.txt
     border
WZZEDOW molstype {
    num rows 10
     textblock
                   moistype, 0,0
TENTELOCK hefdess (
    am_columns
                 10
35
     filename
                   tetblk/huminp.txt
```

```
VARIABLE descripe {
foundin "VARCHAR descripe.arr"
type #FRIED
foundt 4-45s
    DATUM desctype (
       Sam corames
                        77
       verieble
                        desctype, 0 ,24 "Effect descriptor type:"
       helpfile
                        help/hunimp.hlp
    )
DATOM structoff (
       num_rows
                       1
77
                        desctype, 0 ,22 *Structural effect on:*
       verieble
       holpfile
                        belp/strimpec.hlp
WINDOW hkeywrdsearch (
   num_rows 19
num_columns 78
                  0, 0, 0,77
                  qualif, 1,30 descrippe, 2, 1, "CALL WCAPITAL &descrippe",
    datum
    detun
                                       "HENVALS",
                                       "CALL hereh011"
   textblock
                  hefdess,
                               6,23
                              3, 1, "CALL VCAPITAL Smoistype",
    deten
                  moistype,
                                       "HEWVALS",
"CALL herch012"
                  expdess,
                               4, 1, "CALL VCAPITAL Sempdess",
                                       "MENVALS",
                 scrollist, 16,23
stsearch, 18, 1, "CALL herch002"
cancech, 18,45, "CALL questup",
"ERF_SCREEN ching"
   testline
   button
   button
     }
SCHEMA PRODUCTOR (
                 hkeywrdsearch, 2,1
majorastical, 18,1
     Window
      window
     border
    declaration for Structural effects keyword outegory search screen
TENTRLOCK sefdesc {
     aum rous
aum columns
filename
                    10
                     35
                      txtblk/strimpec.txt
                      724
WILEDOW
        skeywideearch {
   200 _ 2006
   300_001m
             me 78
   1120
                 0, 0, 0,77
                 qualif,
                               1,30
   detun
                 structeff, 1, 1, "CALL VCAPITAL Sdesctype",
   detu
                                       "MENVALS",
"ADD WINDOW moistype 7 24"
   testblock
                 safdasa,
                                5,23
   deter
                 moistype,
                                2, 1, "REMOVE_WINDOW",
                                       "CALL VCAPITAL amoistype",
                                       "MENVALS",
                                        "ADD_WINDOW exprtype 7 24",
```

```
"CALL damy"
                ezpássa,
                           3, 1, "CALL VCAPITAL Semplese",
                                     "HENVALS",
                serolltst, 15,23
                stsearch, 17, 1, "CALL dumy"
   button
                 essect,
                            17,45, "CALL quetup",
                                    THE POPULATION COLLEGE
SCHOOL SPOALS (
                skeywrdsearch, 2,1
mejorastica3, 18,1
     window
      window
     border
                774
- Declaration for Noise & Sonic-boom effects keyword ontegory search screen
} Deebda EDOLETEET
                   10
35
     NEW TOWN
     Sum_columns
filename
                    txtblk/sbimpec.txt
     border
                    724
 VARIABLE noiseff {
type STRING
t-45s
   ______format
- DATUM moiseff (
      SWOZ_SWE
        aum_columns
                       77
        variable
                       moiseff, 0 ,32 "Noise and somic boom effect om:"
 "Noise and sonic
helpfile help/nohelp.hlp
WENDOW shkeywrdsearch {
    aum rows 18
aum columns 78
     1120
                 0,0, 0,77
      deten
                  moiseff,
                                1,1, "CALL damy"
                   qualif,
                                 1,30 - ar 2/5 Added
      istum
textblock
                   sbdesd,
                                5, 23
                               2,1, "CALL dummy"
3,1, "CALL dummy"
      detun
                   moistype,
      detun
                   expdese,
                  notimplement,
     testline
                                          5.20
                  scrolltxt, 15,23
stsearch, 17, 1, "CALL dumny"
cencsch, 17,45, "CALL quetup",
      tertline
      button
      bettoe
                                       .MEM SCHOOL OPTOG.
    }
SCHEEN akeysroh (
    title
               "MOISE AND SOUTH-BOOM REVECTS RETRORD SEARCH"
     window
                sbkeywrdsearch, 2,1
      Window
                majoraction3, 18,1
    border
                YES
- Declaration for LBGISLATIVE effects keyword category search screen
-WINDOW lkeywrdsearch (
     AWA_FOWS
                    10
     awn_oolunas
                  70
     1120
                  0,0, 0,77
      detus
                   moiseff,
                                1,1, "CALL downy"
                               1,30 " ar 2/5 Added
2,1, "CALL dummy"
3,1, "CALL dummy"
5,23
     detus
                   qualif,
      detun
                   moistype,
      detun
                   expdese,
      textblock
                  sbdesc,
     textline
                   andy,
                                  5,20
                   scrolltst, 15,23
     textline
```

```
steerch, 17,1
cmacsch, 17,45, "CALL quetup",
"MER_SCREEN dbing"
          button
    -SCREEN likeywrdscheureen (
- title "LEGISLATIVE EFFECTS SEARCE BY RETROPO"
         wiadow
                       lkeywrdsearch, 2,1
        window
                       majoractical, 18,1
              CITATION DISPLAY SCREEN
   TEXTLINE shwreviewtzt ("SHOW REVIEW")
   BUTTON shareview {
        hum rows 1
num_columns 15
textline shwreviewtxt, 0,2
helpfile help/ditation.hlp
  TEXTLINE Chinquirtat ("DATABASE DEGUTRY SCHEDE")
  BOTTON dbinquir (
        hum rows 1
hum columns 25
        testline dbinquirtut, 0, 2
helpfile help/citation.hlp
  TEXTLINE altseltzt ("Alternative selections you can now make:")
  TENTLINE rescopetat ("RESCOPE SEARCE")
  BOZZON Zescope (
       AUR_rows 1
RWM_columns 27
       textline rescopetat, 0, 2
helpfile help/rescope.hlp
  TENTRLOCK sipistf (
       fileame birpit/sipintf.bpl
hum rows 16
num columns 76
border YES
       border
 WINDOW shunkteit (
aum_rows 17
aum_columns 78
        textblock sipintf, 1,1
TEXTLES shwartcitzt ("Show mext citation")
 BUTTON shwantait (
    hum_rows 1
    atm_columns 22
   textline shwartcitxt, 0, 2
helpfile help/mobelp.hlp
TEXTLINE shweritrvixt ("Show critical review (if any)")
BOTTOM shwaritry (
    AME columns 35
     textline shwcritrvtxt, 0, 2
helpfile help/nohelp.hlp
TEXTLINE shwabstrt ("Show abstract (if any)")
BUITON shwabet (
```

```
ME TOWN
         num_columns 24
         textline shumbstat, 0, 2
helpfile help/schelp.hlp
   TENTENE prathisoitzt ("Print this citation")
   BUTTON prathisgit (
        men columns 24
        textline prathisuitat, 0, 2
helpfile help/schelp.hlp
  TENTLINE practitat ("PRINT ALL CITATIONS")
  SUFFOR pracit (
      ava_rows 1
ava_columns 27
       textline practixt, 0, 2 helpfile help/sohelp.hip
  WINDOW ditdispastion (
     SAW LOAD
                    2
78
      NAME COLUMNS
                   shwartdit, 0, 1, "CALL shwartdit"
shwabet, 0,40, "CALL deplahet"
shwaritry, 1, 1, "CALL deplarit"
prathisait, 1,40, "CALL dumny"
     button
     betton
     button
button
     button
 VARIABLE shwdesc1 (
    foundin "VARCEAR60 entdess[0].azz"
    type STRING
format 4-60s
     DATOM shwdeedl {
      AND TOWN 1
       variable shwdeed1, 0 ,1
pickable 20
 VARIABLE shwdesd2 {
    foundin "VARCHARGO entdeed[1].arr"
    type STRING
format 4-60s
    DATOM shwdesc2 {
      BUR COLUMN 1
       variable shwdesc2, 0 ,1
pickable 20
VARIABLE shwdesd3 {
    foundin "VARCEAR60 entdesc[2].arr"
    type symmet to see
   ) Epachtic MPEAG
     AME FORE 1
AME COLUMN 65
       variable shudeed3, 0 ,1 pickable 200
VARIABLE shwdesc4 (
Soundin "VARCEARSO entdesc[3].arr"
   type STRING
format 4-60s
   DATUM shwdesc4 (
      ava_columns 65
```

```
variable
                           shwdesa4, 0 ,1
            pickable
                           100
     VARIABLE suitable {
         formdin "VARCEAR suitable.arr"
        type szeme
format tis
        DATOM suitable {
          hum rows 1
           variable suitable, 0, 21
leader "Suitability rating:"
pickable MO
    VARIABLE estrysum (
        foundia "VARCEAR esumb arr"
        type STRING
formet 4-6s
       DATOM entrysum (
          AME FORS 1
          Variable entrynum, 0, 23
Leader "ARAH Citation Eumber:"
pickable 20
          leader
          pickable
   TEXTLES shwtitzt ("Title:")
   WINDOW mitdisplay (
      ham cons
                     78
      detun
                     estrynus,
                                    0, 1
                     estrynum,
suitable,
      detm
      dete
                     when,
                                    0, 60
      detun
                     ditentami, 2, 1
      detu
                     diteutam2, 3, 1
      det
                     ditautama, 4, 1
      detus
                     ditautnem4, 5, 1
     detu
                     ditautness, 6, 1
     detun
                     citautness, 2,42
     detus
                     ditautnem7, 3,42
                     miteutness, 4,42
     detu
     detu
                     ditautness, 5,42
     detus
                    citautnemio, 6,42
     textline
                    shwtitzt, 8, 2
shwdesol, 8, 9
     detu
     detun
                    shwdesc2,
                                  9, 9
    datum
                    shwdesa3,
                                 10, 9
    detus
                    shwdeso4,
                                 11. 9
     button
                               16, 1, "ADD WINDOW shwarteit 4 1" **** ?????? 16, 1, "CALL dummy" 16,28, "CALL meliste",
                     shwartoit,
    button
                    pracit,
    betton
                    rescope,
                                "CALL resourch"
16,60, "CALL molisto",
   betton
                   cenceck,
                                         "CALL quetup",
                                         MEN SCREEN COINQ"
VARIABLE selemit {
   type STRING
format 4-15s
DATOM selerit {
  aum rows 1
   num columns //
variable salcrit, 0 , 22
leader "Selection Criterion: "
              200
  pickable
```

```
WINDOW sitdispheed (
     num rows 3
num columns 78
     SCROOM eitdepl (
    viadov ditdisplan, 5,1
     border YES
  - Abstract/Critical Review Distract School
 TENTLINE sulltest (*
 WINDOW shwattabs {
    hum rows 1
hum columns 15
    textline mulitext, 0,2
 TEXTLES shworitry ("Show another review (if any)")
 BOTTON shwatter (
      RUM_rows 1
      taxtline shwuritry, 0, 2
helpfile help/nobelp.hlp
 WINDOW shwartrey (
   RUM_rows 1
               shwartrev, 0, 1, "REMOVE_WINDOW",
                                     "CALL deplorit"
 PENTLINE abedone ("Done viewing this text") .
 BUZZON abedone (
     Run columns 29
     textline shedone, 0, 2
helpfile help/nohelp.hlp
TEXTRLOGE memotest (
   aum rows 12
num_columns 74
fileneme txtblk/memotxt.txt
WINDOW absdisplay (
  RUE COLUMNS 77
   textline shwitzi, datum shwdesci,
                            1, 2
1, 9
3, 2
   textblock memotext,
                              16,40, "REMOVE_WINDOW",
  button
                 abedone,
                                       "MEN SCREEN citdepl",
                                      "ADD_WINDOW citdispaction 19 1"
  }
SCHOOLS Showabs (
  title "DISPLAY ABSTRACTS AND CRITICAL REVIEWS" window attdisplaced, 2,1 window abedisplay, 5,1 border YES
```

-END OF ASANCIT. SOF

## A.2 Screen Description File for MTR-related Operations

```
-INCLODE STATEMENT FOR THE ASAN typedef DEFINITIONS NUMBER TO
         COMPLIE USUBS.C WITHOUT INCURRING THE WRATE OF THE COMPLIER
  INCLUDE ASSETTIVE.E
  INCLORE ASAM. R
  - DECLARATIONS IN NON-LINEICAL OBJECT SEQUENCE .....
  - Multiple Choice Space
  VARIABLE meldetoo (
    foundia "VARCHARIO depimult[0].arr"
     type STRING
format 4-30s
 DATOM muldetoo (
     num rous 1
     aum_columns 32
    Variable muldat00, 0, 2
pickable . NO
 VARIABLE muldet01 (
    foundin "VARCEAREO deplesatt[1].arr"
    type STRING
format 4-30s
 DATOM muldet01 {
    num_rows 1
    variable muldet01, 0, 2
pickable 20
 VARIABLE suldat02 {
    foundin "VARCEARSO deplumit[2].arr"
    type STRING
format 4-30s
DATOM muldet02 (
   RUM_FORM 2
    num columns 32
   variable muldet02, 0, 2
pickable MO
VARIABLE meldet03 (
   foundin "VARCEARSO deplumit[3].arr"
type STRING
format 4-30s
DAFOM muldat03 {
   aum_rows 1
num_columns 32
   variable muldet03, 0, 2
pickable 20
VARIABLE muldat04 {
   foundin "VARCHAR30 deplault[4].arr"
   type STRING
format 4-30s
DATOM muldet04 {
  num_rows 1
   aum_columns 32
```

```
variable maldet04, 0, 2
pickable MO
  VARIABLE maldatos (
     foundia "VARCHAR30 deplusit[5].arr"
type STRING
found 4-30s
 DATTM suldat05 {
    sum_rows 1
    sum_columns 32
     variable muldat08, 0, 2
pickable MO
 VARIABLE muldatos {
     foundin "VARCHARIO deplement[6].arr"
     type STAING
formet 4-30s
 DATCM muldatos (
    am_rows 1
     awa_oolumas 32
    variable muldet06, 0, 2
pickable 20
 VARIABLE muldet07 {
    foundia "VARCHARIO depleult[7].err"
    type STRING
format 4-30s
 DAFOM muldato7 (
    aum rows 1
aum columns 32
    variable muldat07, 0, 2
pickable 200
 VARIABLE muldatos (
    foundin "VARCHARIO depleuit[8].arr"
    type symmet
formet 4-30s
DATOM muldatos (
    aum_rows 1
    variable muldat08, 0, 2
pickable 20
VARIABLE muldatos (
    foundin "VARCHARIO deplauit[9].arr"
    type STRING
format 4-30s
DATOM muldatos (
   AUR_FOWS 1
    Sum columns 32
   variable muldatos, 0, 2
pickable 30
VARIABLE muldetio {
   foundia "VARCHARSO deplement[10].arr"
   type STRING
formet 4-30s
DATUM muldatio (
   num_columns 32
   variable muldet10, 0, 2
pickable 20
```

```
VARIABLE meldetil {
      foundia "VARCHARSO deplesalt[11].arr"
      type STRING
format 4-30s
   DATOM muldetl1 (
     rum rows 1
rum columns 32
variable muldet11, 0, 2
pickable 80
  VARIABLE muldeti2 (
      foundin "VARCHARIO depimult[12].arr"
      type symmet 4-30s
  DATOM muldet12 {
     num_rows 1
num_columns 32
     variable muldet12, 0, 2
pickable 20
  VARIABLE muldet13 {
     foundin "VARCEAR30 deplault[13].arr"
     type STRING
format 4-30s
  DATOM muldet13
                         €
     atm_rows 1
     awn_oolumns 32
     variable muldet13, 0, 2
pickable 20
 VARIABLE muldet14 (
foundin "VARCHAR30 deplmult[14].err"
     type STRING
formet 4-30s
 DATOM muldet14 (
hum_rows 1
hum_columns 32
    variable muldat14, 0, 2
pickable NO
 VARIABLE muldat15 {
    foundin "VARCEARSO depimult[15].arr"
    type STRING
formet 4-30s
    3
DAFOM muldet15 (
   RWM_FOWS 1
    RUM_COLUMNS 32
   variable muldat15, 0, 2
pickable 20
VARIABLE muldet16 {
   foundin "VARCEARSO deplault[16].arr"
type STRING
format $-30s
DATOM muldet16 {
  num rows 1
num columns 32
variable muldat16, 0, 2
pickable 30
VARIABLE muldet17 {
  foundin "VARCEAR30 deplault[17].arr"
  type STRING
formet 4-30s
```

```
}
   DAFOM muldet17 (
      htm_rows 1
      aum_columns 32
      Variable muldet17, 0, 2
pickable 20
  VARIABLE muldet18 (
      foundin "VARCEARSO depimult[18].arr*
type STRING
formet 4-20s
  DATOM muldet10 (
     Num columns 22
veriable muidatis, 0, 2
pickable 20
  VARIABLE muldet19 (
     foundin "VARCHAR30 depimult[19].arr"
type STRING
found 4-30s
  DATOM muldet19 (
     hum rows 1
hum columns 32
     variable muldet19, 0, 2
pickable 20
  - Planner ettributes
  VARIABLE plearmen (
    foundin "VARCHAR plantness.arr"
type STRING
formet 4-30s
 DATOM plearmen (
    aum_rows 1
aum_columns 50
    variable planram, 0, 20
leader "Your name, please:"
helpfile help/username.hlp
1
VARIABLE password (
type STRING
formet 4-10s
DATCH password (
   AUM POWS 1
    sum columns 55
    variable password, 0 , 23
leader "Please enter password:"
    leader
    belpfile help/password.hlp
- Other General Storage for Varification, Tests, Etc.
VARIABLE noustress (
      foundia "VARCEAR a2bv.arr"
      type
format
                       STRING
                       4-30s
- Noise Source Attributes
VARIABLE sreid {
   foundin "VARCEAR sreid.arr"
   type STRING
format 4-9s
```

```
DATOM street (
       REEL POWE
       arm_columns 60
       variable
                      secid, 0, 21
       leader
                     "Bune of current MER:"
       pickeble
                      100
   VARIABLE srodess (
      foundin "VARCHAR srodess.arr"
type STRING
found 4-54s
   DATOM szods (
         RUE TOWN
          Bum columns 76
          variable srodesc, 0, 22
          leader
                        "Description:
         halpfila
                     help/mtrdesc.hlp
   DANGE Mindes
         PART LOAD
                         1
         BEER COLUMN
                       70
         variable srodesc, 0, 7
leader "Note: "
         leader
         pickable
  VARIABLE detepubl (
foundia "VARCHAR scopdate.arr"
type STRING
formet 4-12s
  VARIABLE schedule (
foundin "VARCHAR srosched.arr"
type STRING
foundt 4-50s
 DATOM schedule (
        EWE POWE 1
        aum_columns 73
        variable schedule, 0, 22
leader "Scheduling activity: "
helpfile help/schdmtr.hlp
 VARIABLE sroorig {
    foundia "VARCHAR STOOTIG.ATT"
type STRING
formet 4-50s
DATOM origint {
    Rum_rows 1
    RUE Columns 73
    veriable
                   ercorig, 0, 22
                   "Originating activity:"
                help/origatr.hlp
   helpfile
~ MTR Attributes
VARIABLE GERETOS (
                                                  - CONSIST Set
   foundia "VARCEAR GEFERTEG. BFT"
   type STRING
formet 43s
DATOM ownertos (
      EME_FOWS
      atm columns
                    27
      variable
                      GERETOG, 0,16
      leader
      helpfile
                     belp/curertos.hlp
```

VARIABLE GERWIdright (

```
type INTROER
formet 42d
   DATOM ourwidingst (
         num_rows 1
num_columns 25
            variable curvidright, 0,16
leader "Width (right): "
helpfile help/curvidrt.hlp
  VARIABLE GETWINIST (
type INTEGER
formet #2d
  DATOM curvidleft (
          BIN TOWN
           num_rows 1
num_columns 25
           variable ourwidleft, 0,16
leader "Width (left): "
helpfile help/ourwidlf.hlp
  VARIABLE ourhighalt (
      foundin "ALTEREC curhighalt.spec"
type STRING
format 49s
 DAFON ourhighalt (
aum_rows 1
aum_columns 27
           variable ourhigheit, 0,16
leader "High altitude: "
helpfile help/ourhielt.hlp
 VARIABLE curlowalt {
    foundin "ALTEFEC curlowalt.spec"
    type STRING
    foundt 49s
 VARIABLE preented (
                                                                      - PREVIOUS Set
     foundin "VARCHAR prearted.arr"
type STRING
formet the
     DATCM presented (

num_rows 1

num_columns 8

Variable presented, 0,4

pickable 80
VARIABLE prewidright (
type INTEGER
formet $2d
 DATOM prewidright (
      RUM TOWN 1
        veriable previdright, 0,4
pickable 20
VARIABLE providinft (
    type inradix
format 42d
    type
DATCM providleft (
aum_rows 1
aum_columns 8
variable providleft, 0,4
pickable 20
```

```
VARIABLE prehighelt (
foundim "ALTEPEC prehighelt.spec"
      type STRING
format the
  DATOM prohighalt (
        awa_columns 10
         variable prehighalt, 0,1
pickable 20
 VARIABLE prelowalt (
foundin "ALTEFEC prelowalt.spec"
type STRING
formet 49s
  - Coordinates
 VARIABLE entlat {
    foundin "COORDINATE ent.let"
    type STRING
    foundt %-13s
                                                         - CURRENT OF MITTER Set
 }
 VARIABLE estlong {
   foundin "COORDIBATE est.lon"
   type #TRING
   foundt | 4-13s
 VARIABLE shwlet {
foundin "COORDINATE show.lst"
type STRING
foundt 4-13s
                                                      - PREVIOUS or SECT Set.
 VARIABLE shwlong {
    foundin "COCKDIENTE show.lon"
    type STRING
      type STRING
format t-13s
 - Mavigation Point Properties
 VARIABLE GREELStyp (
                                                      - CURRENT or ENTER Set
      foundin "VARCHAR curfirtyp.arr"
type STAIRS
formet 412s
)
VARIABLE ourfindist (
    type Directa
format 83d ,
VARIABLE curfixred (
      type INTRONE
format 43d
VARIABLE curfixed (
     foundia "VARCEAR ourfixed.arr"
     type STRING
Sommet 45s
}
VARIABLE GERRAPPE (
     foundin "VARCEAR Gurmavpt.arr"
      type STRING
format 43s
     format
VARIABLE profistyp (
                                                      - PREVIOUS or DISPLAY Set
     foundin "VARCEAR prefixtyp.arr"
     type STRING
format $12s
```

```
VARIABLE prefindist {
    type INTEGER
    formet $34
  VARIABLE prefixred (
type INTROER
formet $34
 VARIABLE prefixed (
foundia "VARCHAR prefixed.arr"
type STRING
foundt 45s
  }
 VARIABLE pressvpt (
foundis "VARCHAR pressvpt.srr"
type STRING
formet 43s
 - Aircraft Parameters
 VARIABLE admine (
foundin "VARCEAR ad name.arr"
type #FFREE
format 4-12s
 DATOM someme {
           aum_columns 25
          variable some, 0, 10
leader "Aircraft:"
pickable 80
 VARIABLE agrams1 {
      foundia "VARCHAR tid.arr"
type STRING
formet 4-12s
 DATCH scame1 {
    sum_rows 1
          ava_columns 35
          variable accemel, 0, 16
leader "Aircraft acce:"
helpfile help/mtraircr.hlp
 - Operations Specification
 VARIABLE day {
foundia "OPERATIONS ope [0].day"
type INTEGER
formet $44
                                                                    - GENERIC OF JAMESHY
 VARIABLE junday {
foundin "OPERATIONS ope[0].day"
type INTRONS
formet 44d
VARIABLE might (
foundin "OFERATIONS ope[0].mite"
type INTEGER
foundt 04d
VARIABLE jennite {
foundin "OFERATIONS ope[0].mite"
type DEFECT
founds $44
VARIABLE febday {
foundia "OPERATIONS ops[1].day"
```

```
type INTEGER
formet $44
  }
  VARIABLE febaite {
         foundin "CFERATIONS ope[1].mite"
type INTRONE
formet 44d
  VARIABLE marday (
foundin "OFFRATIONS ope [2].day"
type DEFECTA
foundt 04d
  VARIABLE mermite (
         foundin "OPERATIONS ope[2].nite"
type INTEGER
formet #44
 VARIABLE sprdny {
foundia "OFERATIONS ope[3].day"
type IFFERER
format 44d
 VARIABLE apraite (
foundia "OFERATIONS ope[3].aite"
type INTEGER
formet 44d
 VARIABLE mayday {
foundin "OPERATIONS ope[4].day"
type INTRODE
formet 64d
 }
 }
 VARIABLE junday (
foundia "OPERATIONS ope[5].day"
type INTROUR
formet #44
VARIABLE jumnite (
foundin "OFERATIONS ope[5].mite"
type INTRACER
format 44d
 VARIABLE julday (
foundin "OFERATIONS ops[6].day"
type DEFECER
formet 04d
VARIABLE julaite (
       foundin "OPERATIONS ops[6].nite"
type DEFRORR
format $44
}
VARIABLE sugday {
foundin "OPERATIONS ope[7].day"
type INTEGER
formet 44d
}
VARIABLE augnite (
      foundin "OPERATIONS ope[7].mite"
type INTEGER
formet 44d
}
```

```
VARIABLE sepday (
foundle "OPERATIONS ops[8].day"
type INTEGER
foundt 44d
 VARIABLE sepaits {
foundin "OFERNIONS ops[8].nite"
type DEFICER
format 04d
 }
 VARIABLE octday {
   foundin "OPERATIONS ope[9].day"
        type IPTECER
                         14d
 VARIABLE octaite {
foundin "CPERATIONS ope[9].mite"
type DEFIGER
foundt 44d
 VARIABLE movdey {
foundia "OPERATIONS ope[10].day"
type INTEGER
foundt $44
 VARIABLE movaite {
       foundia "OFERATIONS ops[10].nite"
type INTEGER
format 44d
 VARIABLE decday (
foundin "OPERATIONS ope[11].day"
type INTEGER
formet $44
 }
 VARIABLE decuite (
foundin "OPERATIONS ope[11].aite"
       type INTEGER formet $4d
 - Missions
 VARIABLE missens (
foundis "VARCEAR missishl.arr"
    type STRING
formet 4-7s
 DATCH missemel (
        BWE FOUR
         aum_oolumns 34
         variable misses, 0, 26
        leeder
                         "Name of Gurrent mission: "
        pickable
                       200
VARIABLE noumissume (
foundin "VARCEAR did.arr"
        type
format
                         STRING
                         4-7e
DATOM mismone {
        BWR_FOWS
        RUE COlumns 24
        variable
        leader
                         "Mission name:"
        helpfile
                       help/missneme.hlp
TEXTLEM select (
"Right now you can type ? for help, <CTRL> C to quit, or nove the oursor")
TEXTLINE eavess {"COMPOUT AN ENVIRONMENTAL ASSESSMENT" }
```

```
THEFLUM househp ("FERFORM DATABASE HOUSEKEEPING")
THEFLUM introtat ("VIEW GENERAL INFORMATION ABOUT THIS PROGRAM")
  THEFLUM selectistri ("Select aircraft and mission for MTR")
THEFLUM modeumstrist ("Modify current MTR")
  TEXTLES object the ("Select another MTR")
  TEXTLEMS next nevertet ("Enter next nevigation point")
  TENTLINE ourpretst
   ( "CURRENT
                   PREVIOUS
                                                     CORRECT
                                                                   PREVIOUS" }
  TEXTLES curpretet3
  TEXTLINE conceletrint ("Cancel this MIR data entry")
  TEXTLEM seventrint ("Seve this MER")
  TEXTLIRE showmoremetrizi ("Show more MER names (if any)")
  TEXTLINE recalistrict ("Recall one of the following MERS:")
TEXTLINE statement ("Start new MER")
TEXTLINE definement ("Enter route maypoints")
  TEXTLINE actst ("Aircraft:")
  TEXTLEM opensonist ("Operations are seasonal")
  TEXTLINE opthraystat ("Operations are even throughout year")
  TEXTLINE instrict ("Please enter day and might operations by mouth")
 TEXTLUE depaits ("DAY SIGST DAY NIGST DAY NIGST")
TEXTLUE instrict ("Please exter deptime and might operations per month")
TEXTLUE canonistst ("Abandon this mission")
TEXTLUE extensectst ("Start new mission")
  TEXTRLOCK introtat (
      num_rows 14
      border YES
filename txtblk/intro.txt
 MOTTOW assessment (
     FIRE LOAD
     aum_columns 36
     textline envass, 0, 2
helpfile help/assess.hlp
 MOTTON abgests (
     NOT TOWN
    ava columns 35
     textline
                       chgmtrtxt, 0, 2
                    help/selstr.hlp
    helpfile
 MUTTON housekeeping (
    Par Lone
    aum_rows 1
aum columns 33
    textline housekp, 0, 2
helpfile help/housekp.hlp
BOTTOM introtat (
    AWE TOWN
    hum_columns 56
    textline
    textline introtxt, 0, 2
helpfile help/introhel.hlp
BUTTOM modomamtr (
   EME_FOWS 1
    atm columns 35
   textline modeumstrtxt, 0, 2
helpfile help/mohelp.hlp
        - so help here our this button doesn't do maything
WINDOW password {
    NEET LOAD
    num_columns 60
    datum password, 0, 5
TEXTLEME tititat
{"Developed for Hoise and Sonic Boom Impact Technology Program"}
```

```
TEXTLES tit2tst ("under V.S. Air Force Contract F33615-86-G-0530")
   TEXTLINE titStat ("by BBW Laboratories, Inc.")
   TEXTLES tit4tst ("February, 1986")
   TEXTLINE titStat
   ("Unreleased demonstration of Prototype Version...Hot for General Dee")
  TRITLINE diafo ("Done viewing general information on ASAS")
  BUTTON diafo (
      REAL POWS 1
       textline dinfo, 0, 2
  WINDOW introduction (
      num_rows 15
num_columns 78
      textblock introtut, 0, 1
button diafo, 14, 1, "REMOVE WINDOW"
      button
  WINDOW introwindow1 {
      2000
      REE COLUMN 78
      1120
                    1, 0, 1, 77
      title
                    "ASSESSMENT SYSTEM FOR AIRCRAFT HOLSE (ASAM) "
      textline
                    tititzt, 2, 9
      tertline
                    tit2tmt, 3, 16
      tertline
                    tit3tst, 4, 27
      textline
                    titétat, 5, 32
titétat, 7, 6
      textline
      محلة
                    8,0, 8,77
      det m
                    plantam, 12, 1,
                    CALL VCAPITAL Spinnson
                    "ADD_WINDOW password 15 3",
                    "REMOVE WINDOW",
                    "CALL pucheck"
  )
 WEMDOW introvindow2 (
     Tam Tone
     arm_columns 78
     معند
                   5, 0, 5, 77
                  select, 6, 3
introduct, 7,15, "ADD_WINDOW introduction 3 1"
assessment, 8, 2, "CALL peprobet"
housekeeping, 8,44, "CALL pedbhack"
     textline
     button
     button
     button
 SCREEN firstscreen (
    titlescreen TES
    border
                  YES
    title
                  "ASSESSMENT SYSTEM FOR AIRCRAFT HOISE (ASAE)"
                 introvindow1, 1,1
introvindow2, 13,1
    Window
    Window
BUTTON OPERAGE (
   num columns 41
   tartline opsessontri, 1, 3
border TES
   helpfile help/weblops.hlp
BUTTON operendy (
   MADE TOWN
   BUR Columns 41
   textline opthraystat, 1, 3
border YES
   helpfile help/evenops.hlp
BOTTON mulbutoo {
  NUM_rows 1
  NUM COLUMN 1
  belpfile belp/mulbet.blp
```

```
BOTTON mulbetol (
    num rows 1
     helpfile help/mulbut.hlp
  MOTTOW mulbut02 (
     num_rows 1
num_columns 1
helpfile help/mulbut.hlp
  NOTZON melbetos (
     aua_rows 1
     awa columns 1
     helpfile help/mulbut.hlp
  MOTTOW mulbet04 (
    num rows 1
     helpfile help/mulbut.hlp
  NOTION malbatos (
    num_columns 1
    helpfile help/mulbut.hlp
 NOTICE mulbetos {
    aum_rows · 1
    atm_columns 1
    helpfile help/subst.hlp
 NOTION mulbut07 {
    num_rows 1
num_columns 1
helpfile help/mulbut.hlp
 NOTION malbatos (
   num_rows 1
num_columns 1
helpfile help/nulbut.hlp
 NOTICE mulbetos (
   aum_rows 1
aum_columns 1
    helpfile help/mulbut.hlp
MOTTOW malbet10 {
   aum_rows 1
   num columns 2
   helpfile help/mulbut.hlp
NOTION mulbet11 (
   ETEL_TOWN 1
   ava_columns 2
   helpfile help/mulbut.hlp
NOTICE malbet12 (
  htm_rows 1
   helpfile help/mulbut.hlp
MOTTON malbut13 (
  amm_rows 1
  RUM Columns 2
  belpfile belp/mulbut.hlp
BUTTOW mulbet14 {
  aum_rows 1
  ava columns 2
  helpfile help/mulbut.hlp
```

```
MOTTOW mulbut15 (
    aum_rows 1
    aum columns 2
    helpfile help/mulbut.hlp
 NOTICE malbet16 {
   num_rows 1
num_columns 2
    helpfile help/mulbut.hlp
 NOTICE malbet17 (
    aum_rows 1
aum_columns 2
    helpfile help/mulbut.hlp
 SOFFON melbetis (
    REAL FORM 1
    num_columns 2
    helpfile help/mulbut.hlp
 NOTION malbet19 (
   num_rows 1
    helpfile help/mulbut.hlp
         Declarations for MTR DATA MATER SCREEN
NOTICE stanuis (
      am_columns 40
     textline streemistrt, 0 ,2
helpfile help/streemis.hlp
MOTTON selectis (
     Man Lone
      am columns
textline
                     40
                   selecmistat, 0,2
help/selecm.hlp
      helpfile
TEXTLEM spectruistat ("Specify new mission")
BUTTON spectatels (
     RUE COME 1
     textline specumistri, 0, 2
helpfile help/neumiss.hlp
WINDOW strdatestry (
   8 swor_mra
   awa_columns 78
               object, 0, 2, "CALL pedagatr"
selecais, 2, 2, "CALL pentrais"
modoumetr, 4, 2, "CALL dumny"
spectrumis, 6, 2, "ADD_WINDOW nermisms 13 3",
"UDDATE DATME missams",
   button
              abgatr,
   button
   button
   bettoe
                                    "CALL WEEREAMI Screen Window Datum Button",
                                   "CALL stropy oldscreen Screen",
   }
WINDOW stramoon (
  rows 3
aum columns 78
line
                 2,0, 2,77
   detun
                 mtram, 0, 2
mtrdess, 1, 2
WINDOW mtrassocal {
```

)

```
78
                       3,0, 3,77
                      mtraam, 0, 2
mtrdesc, 1, 2
                       missemel, 2, 2
           Declarations for DEFINE/MODIFY SCREEN (defmodstreamen)
  DAFOM prelowalt {
       num columns 10
        variable prelocalt, 0,1
       pickable
  DATUM curlowalt (
       ETEL_POWE
        ava_columns 29
        variable curlowalt, 0, 16
leader "Low altitude: "
       helpfile help/curlcalt.hlp
  DATUM prefixtype (
      RUE COLUMN 1
        Variable prefixtyp, 0,0
pickeble 20
       pickable
 DATOM curfixtype (
      PART TORK
       ATE COLUMN 24
       variable curfixtyp, 0,11
leader "Fix type:
       helpfile
                       help/curfixtp.hlp
     Declarations for entering coordinates
 DATOM entlong {
       Num_rows 1
      aum_columns 25
variable entlong, 0 ,11
lender "Longitude:"
helpfile help/merkmap.hlp
DATOM entlet (
     num rowe 1
num columns 25
variable entlat, 0 ,11
leader "Letitude: "
helpfile help/merimep.hlp
          Declarations for show coordinates
DATUM shwlet (
      RME FOWS 1
      num columns
                       14
      Variable shwlat, 0 ,0
pickable NO
helpfile help/combmap.hlp
  }
DATCH shwlong (
      ava_columns 14
     variable shwlong, 0 ,0
pickable NO
helpfile help/combmap.hlp
```

```
DATOM profindist (
        Par Lond
         num_columns 8
         verieble
                    prefindist, 0,4
        pickable
   DATOM ourfindist (
         SAM_LOAM
         ham_columns
                       26
                       Garfindist, 0,16
"Fix distance: "
         Variable
        leader
        belpfile
                     help/ourfirdi.hlp
  DATM prefixed (
        awa columns
        variable prefixed, 0,4
pickeble 20
        pickable
   DATOM GERTLERED (
        THE TOWN
        num_columns
                      26
        variable curfixred, 0,16
        leader
                      "Fix redial:
        helpfile
                    help/ourfirms.hlp
  DATOM profixed (
        ava rows 1
       variable prefixed, 0,4
pickable 20
  DATOM ourfixed (
       hum rows 1
        ham columns 26
        variable
                      ourfixed, 0,16 "Fix ID:
       helpfile
                      help/ourfixed.hlp
 DATCM preseypt (
      NUM rows 1
NUM columns 10
Variable pressvpt, 0,4
pickable 20
 DATOM GERMANPE (
      aum_rows 1
aum_columns 24
      variable curnavpt, 0, 16
leader "Env. Foint: "
helpfile help/curnavpt.hlp
 DATCM nountres (
      PAR LOAM
      20010000
                      40
                      nountrem, 0,7
      verieble
      leader
      helpfile
                     help/stretres.hlp
MUTTON mestasept (
     hum rows 1
     textline nextsavptst, 0, 2
helpfile help/axtsav.hlp
TEXTLINE retairest the ("Continue without selecting MER")
TEXTLINE retairest ("Continue without greating mission")
TEXTLINE retairest ("Save mission in database")
TEXTLES retnomis
                          ("Continue without selecting mission")
```

المساحة الماساليين

```
SUFFOW retutrest (
     EWE TOWN
     num_columns 40
     textline
                  retmirentat, 0 ,2
     bolpfile
                  help/noneuntr.hlp
 BUTTOW retutreet1 (
    ZWZ_FORE
    aum_columns 40
    textiine
                  retmiremint1, 0 , 2
    belpfile
                help/nomiss.hlp
 SUFFOR retutrent2 (
    RUM_FOUR
    aum_columns 30
             retmtreatxt2, 0 , 2
help/saveniss.hlp
    textiine
    helpfile
 NOTTON retacals (
    atm_rows 1
    num columns 40
    textline retnomis, 0 , 2 helpfile help/nonewmis.hlp
 WINDOW definedate {
       ETE FOWS
                     15
       num_columns
                     78
       textline
                    ourprotet,
                                  0, 16
       textline
                     ourpretzt3,
       detus
                    ournevpt,
                                   2, 1, "CALL VCAPITAL &GERRAUPT",
                                           "BENVALS"
                                  2. 26
                    prenavpt,
                                  3, 1, "CALL VCADITAL Sourfixid",
       detun
                    curfixid,
                                           "MENVALS"
       detu
                    prefixid,
                                  3, 26
       detun
                    ourfixed,
                                  4, 1
       detun
                    prefixed,
                                  4, 26
       detu
                    ourfirdist,
                                  5. 1
                    profindist,
       deten
                                  5. 26
       detu
                    entlet,
                                  2, 38, "CMLL lat2ded Seat",
                                           "MEWVALE"
       detun
                    shwlat,
                                  2, 64
       detre
                    emtlong,
                                  3, 38, "CALL lon2ded Sent",
                                           "MENVALS"
       detus
                    shwlong,
                                  3, 64
                    corfixtype,
       detun
                                  4, 30, "CALL VCAPITAL SCHIFFIXTYP",
                                           "MEWVALS"
       detu
                   prefixtype,
                                  7, 2, "CRLL altided &curlowalt",
      dete
                    ourlowalt,
                                          "DEEVALE"
                                  7, 26
      dete
                   prelowalt,
      detun
                    ourhighelt,
                                 8, 2, "CALL alt2dec Sourhighalt",
                                           "MEWVALS"
      detun
                   prohighalt,
                                 0, 26
      detu
                   ourwidleft,
                                 9, 2
9, 27
      detu
                   providleft,
                    curwidright, 10, 2
      detu
      detun
                   prewidright, 10, 27
      deten
                    curartec,
                                 11, 2, "CALL VCAPITAL Sourertoo",
                                           "MENVALE"
      detu
                   preertoo, 11, 26
mentaevpt, 13, 2, "CALL astmirpt"
                                11, 26
      bettos
    SEE_FOUR 1
BUTTON cancelmir (
    taxtline cencelatrixt, 0, 2
helpfile help/candistr.hip
MUTTON SEVENTY (
    AUM_rows 1
    ave columns
                  20
    tartline savestrint, 0, 2
helpfile help/savestr.hlp
```

```
)
 WINDOW straction
       awa_colu
        1120
                       0,0, 0,77
       button
                            mtr, 1, 2, "CALL seventr"
       button
                       conceintr, 1,40, "CALL concert"
 SCREEN defendatrucreen (
title "DEFINE/MODIFY MER"
                  mtraenoom, 2, 1
defmodstr, 6, 1
       window
       window
       window
                  mtraction, 20, 1
      border
                  774
                  Declarations for SELECT ANOTHER MER SCREEN
     num_columns 27
taxtline showncountrixt, 0,2
helpfile help/shumtrs.hlp
NOTICE stansar {
    sum_rows 1
    sum_columns 40
      textline stnesstrtxt, 0 ,2
helpfile help/startstr.hlp
 WINDOW Bountres (
   num_rows 1
num_columns 40
                  nountrum, 0, 2
WINDOW streomistr {
   REAL POWE
                   4
   nam_contame
                   stneumtr, 1, 2,
                   "ADD_WINDOW novembran 6 3",
                   "UPDATE DATCH newstree",
                   "CALL penuntra a2bv.arr"
                  shownerstr, 1, 41, "CALL mebunch"
   textline
                   recellmitrist, 3, 2
WINDOW Gurdet {
    datum muldatoo, o, o
    detum muldat01, 1, 0
    detum muldet02, 2, 0
    detum muldet03, 3, 0
    detum muldat04, 4, 0
    detum muldet05, 5, 0
    datum muldatos, s, o
    datum muldato7, 7, 0
    datum muldatos, s, o
    datum muldatos, s, o
WINDOW gurdet1 (
    num_rows 10
    datum muldet10, 0, 0
    detum muldet11, 1, 0
    detum muldet12, 2, 0
    datum muldati3, 3, 0
    datum muldet14, 4, 0
    datum muldet15, 5, 0
    datum muldatis, 6, 0
    datum muldet17, 7, 0
   datum muldatis, s, o datum muldatis, s, o
```

```
WINDOW Gummtrhet (
        ___rows 10
        betton mulbetoo, 0, 0, "CALL MIRCORN deplanat[0].arm"
        bettos malbet01, 1, 0, "CALL MERGORA deplanit[1].arr"
bettos malbet02, 2, 0, "CALL MERGORA deplanit[2].arr"
        betton mulbet03, 3, 0, "CALL MFROOM deplault[3].arr"
betton mulbet04, 4, 0, "CALL MFROOM deplault[4].arr"
        button malbut05, 5, 0, "CALL MFRooms deplanat[5].arr"
        betton mulbutos, 6, 0, "CALL MIRcoom deplant[6].arr"
        button malbut07, 7, 0, "CALL MTRooms deplault[7].arr"
       bettom mulbutos, s, o, "CALL MIRCORN deplault[s].arr"
       button melbut09, 9, 0, "CALL MIROCON deploralt[9].arr"
   WINDOW
             ouzstabeti (
       hum_rows 12
hum_columns 2
       button malbet10,
                             0, 0, "CALL MIROCAL deplet[10].arr"
1, 0, "CALL MIROCAL deplet[11].arr"
       button mulbut11,
       button malbut12,
                             2, 0, *GALL MIRCOGN deplesalt[12].arr
       button mulbut13,
                              3, 0, "CALL MIROORN deplement [13] arr"
       betton mulbut14,
                              4, 0, "CALL MIROCOR deplault[14].arr"
       betton malbut15,
                              5, 0, "CALL MIROCON deplault [15] .arr"
       button malbut16,
                              6, 0, "CALL MIRCORN deplumit[16] .arr"
       button malbut17, 7, 0, "CALL MTRooms deplault[17].arr"
button mulbut18, 8, 0, "CALL MTRooms deplault[18].arr"
button mulbut19, 9, 0, "CALL MTRooms deplault[19].arr"
  WEMPOW returnet {
       BWE FOUR 1
       awa columns 45
                  retutrent, 0, 1, "CALL melista",
                                        "CALL postront"
      }
  SCHEM chgountrecreen (
       title "SELECT ANOTHER MER"
       window strassoon, 2, 1
       window strecalmtr, 5, 1
       Window oursetzbut, 9, 3
       window oursetribut1, 9,42
       window
               ourdet, 9, 5
       window gurdati, 9,44
       window retmirent, 20, 2
       border YES
                          Declarations for new mir name
 DATOM datepubl (
      aum_rows 1
       num_columns 50
      Variable detepubl, 0, 22
leader "Date of publication: "
      helpfile help/datepubl.hlp
BUTTON definents (
     num_columns 40
tertline definemtrtxt, 0 ,2
helpfile help/getmep.hlp
WINDOW defauntr
      BUR FOWS
       aws_columns
                      77
       detu
                       origatr,
                                    0, 1
                       schedule, 2, 1
       detus
       detun
                       srods,
                                    4, 1
       detu
                      datepubl, 6, 1
definemtr, 8, 1,
       button
                                                    "CALL estatrpt"
SCREEN mirdefinesureen (
     title "MTR DEFINITION"
```

```
3,1
7,2
                   defauntr,
                  mtractice,
                                   20,1
                    Declarations for MISSIGN REQUIREMENTS WINDOW
  VARIABLE se_in_form (
type DFFERE
       type
format
                     42d
       lowlimit
                     1
       wlimit
                    16
       defeult
                    2
  VARIABLE perstuat (
      foundin
                "VARCEAR PT_PWT_E.AFT"
      type
format
                    FIRING
                   4-60
 DATOM purstuat (
     awa_columns
                 10
                 perstunt, 0, 2
"["
"]"
     veriable
     leeder
    trailer
    pickable
    •
 VARIABLE mistype (
type STRING
formet 0-1s
 DATOM mistypel (
     num rows 1
      variable mistype, 0, 14
leader "Mission type:"
                 help/mistype.hlp
      helpfile
 DATOM mistype {
      num_columns 18
                   mistype, 0, 14
"Mission type:"
      veriable
      leader
     pickable
                    10
VARIABLE presitiev (
foundia "ALTEFEC prelomeit.sped"
     type
formet
                   STRING
                    490
DATEM prosition (
     aum columns 10
     Run_rows
     variable presitiev, 0, 0 pickable NO
VARIABLE oursitles
     foundin "ALTSFEC Gurlowelt.spec"
     type
formet
                   STRING
                   490
DATEM curaltiev (
     RUM FOWS
     verieble
                   Curaltlev, 0, 11 *Alt: *
     leeder
    helpfile
               help/altmtr.hlp
VARIABLE ad pro pur (
```

```
type
format
                       ₹12.31£
        lowlimit
                      0.0
        eplinit
                      4000.0
        default
                      100.0
  DATCM prepurset {
    Rum rows 1
    Rum columns 1
                  1
ns 12
       variable ac pre pur, 0, 0 pickable 20
  ANTINETS sc car bar {
       type
       formet
                      410.31£
       lowlimit
                      0.0
       welimit
                      4000.0
       default
                      100.0
  DATOM ourpwreat (
       RUM FOWS
       NUM_columns
                    21
       veriable
                  ac_car_pwr, 0, 11
       leader
                  help/pwrspec.hlp
       helpfile
  VARIABLE ad pre_spd (
     type :
                     434
 DATUM prespeed (
      aum rows 1
aum columns 9
      variable ac pre spd, 0, 0 trailer "Ergs" pickable 20
      pickable
 VARIABLE sc_our_spd {
    formet
    lowlimit
                   100
    uplimit
                   600
    default
                   450
 DATEM curspeed {
     aum_columns 21
variable
                    ac_our_spd, 0, 11 "Speed: "
      leeder
      trailer
                    "Ers"
      helpfile
                 help/mtrspeed.hlp
DATEM pressypt1 (
                    1
     num_columns 1
    variable premavpt, 0,0
pickable 20
DATOM Guracypt1 (
     Nama Tours
                  21
     aum columns
                  Gurnavpt, 0, 11
"Mavpoint:"
      verieble
     leeder
     helpfile help/nevnee.hlp
TEXTLES canciparate ("Cancel data entry for this mission")
BUTTON cancipara (
    num_rows 1
num_columns 15
tertline canciperatut, 0, 2
helpfile help/cancimer.hip
```

```
TEXTLES saveparetat ("Save this mission's data")
   BOTTON savepara (
        BUR TOWN
        ave columns
                       30
                         saveperstat, 0, 2
                      help/saveniss.hlp
        helpfile
  TEXTLES operdatatut ("Ready to enter operational data")
  BUTTON operdate (
      BUR TOWN
        EWE_COLUMN
                       33
       taxtline operdstatxt, 0, 2
helpfile help/entrope.hlp
  TRATELOCK mir (
num_rows 10
num_columns 38
filesume txt
                  tatblk/str.tat
 TEXTLINE Etrlabeltzt ("MAY FIX
                                                 FIX TYPE
                                                                   WADTE" )
 TRETLINE strict ("REFERENCE INFORMATION FOR DATA SHIRY")
  TRATLINE CUIPTOLELL ("CORRERT PREVIOUS")
  TEXTLINE Curpretat2 ("----")
 WINDOW opentry (
    BUM_TOWN
    num_rows 10
num_columns 78
button operandy, 2, 19, "CALL penddops 1"
button operanson, 7, 19, "CALL penddops 12"
                        10
 WIMDOW fitpers (
       aum_rows 17
        det un
                       mistype,
somme,
                                      0, 2
1, 2
0,40
                       strtzt,
        textline
        textline
                       mtrlabeltzt, 2,38
        textline
                       Gurpretzti, 3,14
Gurpretzti, 4,14
Gurpretzti, 5,1, "CALL VCAPITAL &Gurmavpt",
        textline
        deton
                                             "MENVALS"
        detu
                       presavpt1, 5,25
        dete
                       prespeed,
                                     7, 1
        detun
                                     7,25
        detun
                       curpurset,
                                     9, 2
                      preparate, 9,22
perstuat, 10, 15
curaltlev, 12, 2, "CALL alt2dec Scurlowalt",
        detun
       detun
       detun
                                              "MEWVALE"
       dete
                       presitiev, :
                                     12,23
       box
                       3, 30, 3, 76
15, 0, 15, 77
       Line
       1420
                      nerthaupt, 14, 2, "CALL netmispt"
operdata, 16, 2, "MEW_SCREEN opentry"
cancipara, 16,40, "CALL cancais"
       button
       button
       button
       textblock
                      mtr,
                                    4,38
SCHOOL Mirflt (
     title
Window
                  "FLIGHT PARAMETER MITRY"
                 mtramoom, 2,1
      window
                 fitpera,
                                   5.1
                 TES
SCREEN opening {
    **FLIGHT OPERATION DATA ENTRY FOR MER-
                 mtrnamoom, 3,1
     window
                 opentry,
```

```
Declarations for seasonal MTR and REQUIREMENTS
 DATOM domite (
      num rows 1
num columns 6
      variable
                    decaite, 0, 0
      helpfile
                    help/mathaite.hlp
 DATM decday {
      num columns
Turisble
                   10
                    deodey, 0, 5
      leeder
      helpfile
                    help/mathday.hlp
 DATTM movaite (
      AUR_rows 1
      num_columns 6
      variable
                    novmite, 0, 0
help/mathmite.hlp
      belpfile
   }
 DATOM novday (
     num rows 1
      amm_oolumns
                   10
                    movday, 0, 5
      variable
                    "WOV: "
      leader
                    help/mnthday.hlp
     helpfile
 DATOM ostaite {
     num_rows 1
      Natiople
                  •
                    octaite, 0, 0
     helpfile
                    help/mathaite.hlp
 DATUM outday (
     aum columns
variable
                   10
                    octday, 0, 5
     leeder
     belpfile
                    help/methday.hlp
DATTM sepaite {
     awa rows 1
     veriable
                   sepaite, 0, 0
     helpfile
                   help/mathaite.hlp
DATTM sepday (
aum_rows 1
aum_columns 10
     veriable
                   sepday, 0, 5
     leeder
     belpfile
                   help/mathday.hlp
DATOM emgmite (
     num_rows 1
num_columns 6
     variable
                   augmite, 0, 0
help/mathmite.hlp
     helpfile
DATOM supday (
num_rows 1
     num_columns 10
     variable
                   augday, 0, 5
     leader
     helpfile
                   help/mathday.hlp
DATOM julnite (
num_rows 1
```

}

```
julnite, 0, 0
help/methnite.hlp
       verieble
       belpfile
 DATUM julday {
       aum_columns
                     10
                      julday, 0, 5
       leader
       helpfile
                      help/mathday.hlp
 DATOM junnite {
    num_rows 1
    num_columns
       variable
                    junnite, 0, 0
belp/mathmite.hlp
       belpfile
 DARTM junday (
num_rows 1
num_columns
variable
                    10
                     junday, 0, 5
"JUN:"
help/mathday.hlp
       leeder
       helpfile
 DATOM maymite (
      num rows 1
num columns 6
       verieble
                     maymite, 0, 0
      helpfile help/mathnite.hlp
 DATCH mayday (
      Bum rows 1
       num_columns 10
      variable mayday, 0, 5
leader "MAY:"
                  help/mathday.hlp
      helpfile
   }
 DATOM apraite (
      awa_rows 1
awa_columns 6
      variable
                    apraite, 0, 0
help/mathaite.hlp
      helpfile
 DATOM sprday (
      num_rows 1
      ava_columns 10
      variable aprday, 0, 5
leader "APR:"
      halpfile
                  help/mathday.hlp
DATOM marmite (
     BUR_rows 1
     num_columns 6
     variable marnite, 0, 0
helpfile help/mnthmite.hlp
DATOM marday (
     RUM_FOWS 1
     aum_columns 10
     variable
                    marday, 0, 5
     leeder
     helpfile
                    help/mathday.hlp
variable febrite, 0, 0
belpfile help/mnthnite.hlp
  )
DATOM febday (
   BUR FOWS 1
     aum_columns 10
```

```
belpfile
                      help/mathday.hlp
  DATOM jammite (
       num_rows 1
       num columns
                      •
                      jamaite, 0, 0
       variable
       helpfile
                      help/mthnite.hlp
 DATTM jenday {
      RUM FOWS 1
                     10
       variable
                      janday, 0, 5
       leeder
       helpfile
                      help/mathday.hlp
 WINDOW month (
      num_rows 14
      aum_columns 77
      textline instrict, 1, 1
textline depaite, 3, 0
detum junday, 4, 1
      detus janday,
detus janaite,
detus febday,
                          4, 12
                           5, 1
       datum febalte,
                          5, 12
       detum marday,
                           6, 1
       detum marmite,
       detum eproey,
                           4,20
       detum apraite,
                           4,31
       detum mayday,
                           5,20
      datum maymite,
                           5,31
      datum junday,
                           6,20
      detum jummite,
                           6, 31
      detun
              julday,
                           4,39
      detum julnite,
                           4,50
      datum augdey,
datum augmite,
                           5.39
                          5,50
      detum sepday,
detum sepaite,
                           6,39
                           6,50
      datum octday,
                           4,58
      detum octaite,
                          4, 69
      detun
                          5,50
             novday,
      datum novaite,
                          5, 69
      datum deoday,
      detum decaite,
                          6, 69
     button savepara, 13, 2, "CALL savemis"
button candipara, 13,40, "CALL canonis"
       Declarations for new SELECT AIRCRAFT AND MISSION FOR MIR
TEXTLUE shunoremistri ("Show more mission names (if any)")
BOTTOW shunoremis (
     MM_FOWS
     NAME OF PARSE
                    30
     textline
                      shemoremistrt, 0, 2
     helpfile
                   help/slamiss.hlp
TEXTLES receives ("Receil one of the following missions")
WENDOW mountains (
   2000
   aum_rows 1
aum_columns 40
                 misseme, 0, 2
WINDOW Curmies (
    num rows 16
                          1, 2, "ADD_WINDOW newmishm 7 3", "UPDATE_DATOM mishame",
                                 "CALL penumien old.arr"
   button shumoremis, 1, 40, "CALL sebunch"
    textline receimis, 3, 2
```

veriable

leeder

febday, 0, 5

```
detum muldetoo.
       detun
             muldat01,
                            5, 4
      det un
             muldet02,
                             6. 4
      deten
             muldet03,
             muldet04,
       det
      datum muldet05.
                             9, 4
       detun
             muldat06,
                            10, 4
      datum muldator,
                            11. 4
      datum muldetos,
                            12. 4
      detun
             muldat09,
                            13, 4
      datum muldatio,
                            4,42
      detum muldet11,
                            5, 42
      detum muldet12.
                            6, 42
      detum muldetil.
                            7,42
      datum muldet14,
                            8.42
      datum muldet15,
                            9.42
      detum muldet16,
                            10.42
      detum muldet17,
                            11,42
      detun
            muldet18,
                            12,42
      datum muldet19,
                            13,42
     button
                            4, 2, "CALL MISCORE deplement[0].arr"
             mulbut00,
     button mulbut01,
                            5, 2, "CALL MISCORE depletit[1].arr"
     button
              mulbet02,
                            6, 2, "CALL MIScoun deplanat[2].arr"
     bettos
              mulbet03.
                            7, 2, "CALL MISCOUR deplemit[3] arr"
8, 2, "CALL MISCOUR deplemit[4] arr"
9, 2, "CALL MISCOUR deplemit[5] arr"
     bettoe
              mulbut04,
      button
              mulbut05,
                            10,2, "CALL MISCORN deplement[6] arr"
     button
             malbutos,
     button
              mulbut07,
                           11, 2, "CALL MISCORN deplault[7].arr"
     bettom mulbetos,
                           12, 2, "CALL MISCORN deplanit[8].arr"
     buttos
             mulbut09,
                           13, 2, "CALL Misconn deplanit[9].arr"
     button
             mulbut10.
                            4,40, "CALL MISCORN deplant [10] .arr
     button
             malbut11,
                            5,40, "CALL MISCORN deplet[11].arr"
     button
             mulbut12,
                            6,40, "CALL MISconn deplault[12].arr"
                            7,40, "CALL MISCORN deplement[13] .arr"
     button
             mulbut13,
     button
             mulbut14,
                            8,40, "CALL MIScoon deplault[14] .arr"
     betton
             malbut15
                            9,40, "CALL MIScomm deplacett[15].arr"
     button
             mulbut16,
                          10,40, "CALL MISCORN deplanit[16].arr"
11,40, "CALL MISCORN deplanit[17].arr"
     button
             mulbut17,
     button mulbutis,
                          12,40, "CALL MIScoun deplanit[18] arr"
     button
             mulbut19,
                          13,40, "CMLL MIScoon deplemit[19] .arr"
     button retnomis,
                          15, 2, "CALL selisto",
                                  "CALL HEN SCREEN oldscreen"
TEXTLES salmisstat ("Salect mission")
 MOTTON selmins (
     AWA_rows
      BUR Columns
                      38
      textline
                      selmisstat, 0, 2
      helpfile
                     help/selmiss.hlp
TEXTURE fitperinftst ("Specify flight persector information")
SUFFOR fitperinf (
     BUR TOWN
     ave_column
     textline
                     fltperinftst, 0, 2
     helpfile
                     help/fitperem.hlp
WENDOW selection (
     ME TOWN
                         15
     am_column
                          77
     datum accessel,
                         1,1, "CALL vfyscatr tid.arr"
     button selmiss, 3,2, "CALL pechanis"
button fitperinf, 5,2, "CALL pentifit"
button retairmenti, 13,2, "CALL communis"
SCHEEK spentrals (
    title
                "SELECT AIRCRAFT AND MISSION FOR MER"
     window
                mtrasmooml, 3,1
     Window
                selection, 7,1
     border
SCHEEN abquie
    title
                "SELECT AIRCRAFT AND MISSION FOR MIR"
    window
                mtrnsmooml, 2,1
                Curaise.
```

```
border
                      125
    " Declarations for Day-Hight window
    DATTM might (
          awa_rows 1
awa_columns 16
                           might, 0, 8
          leader
                            "Might:
          helpfile
                         help/miteops.hlp
   DAROM day (
         BUR FORE
          num_columns 16
          Variable
                           day, 0, €
         leeder
                           "Day: "
         helpfile
                       help/dayops.blp
   WINDOW dayaite (
       Bun rows 12
        amm columns 77
datum acamms, 1, 1
taxtline instrict, 3, 1
datum day, 5, 2
datum night, 5,15
button savepars, 10, 2, "CALL expmops",
"CALL savenis"
        button cancipers, 10,40, "CALL cancais"
border YES
           Declarations for mission specification window
  DATOM sortie (
       NUM FOUR 1
        variable as in form, 0, 35

leader Tumber of aircraft in formation: *
helpfile help/numform.hlp
  VARIABLE misdess (
     foundin "VARCEAR misdesc.arr" "this description is for mission
     type STRING
formet 4-60s
 DATCH misdesci (
    aum_rows 1
aum_columns 76
    variable misdess, 0, 7
leader "Descr:"
pickable 20
 DATTM misdess (
    RUM_COLUMNS 76
    variable misdesc, 0, 14
leader "Description:"
helpfile help/misdesc.hlp
| Lanconaria WOOKIW
   aum_rows 3
aum_columns 76
line 2,0,2,77
datum misname1, 0,2
datum misdesd1, 1,2
WINDOW misspec {
   aum_rows 16
   aum_columns 77
   datum misdesc, 1,1
```

```
datum mistypel, 3,2
datum sortie, 5,2
     betton retmirent1, 14,2, "CALL MEN SCREEN oldscreen"
     button retutrent2, 14,47, "CALL entermis"
 SCREEN misspec {
title "MISSION SPECIFICATION FOR AN MER"
       title
Window
                 mission specifi
misspec, 6,1
TES
 -01-14-88 st will try to fix this file so it wil be only for assessments changes will be commented with date change title on first screen, change lat. long. format lon map control
  -01-17-88 st com't to fix sdf file commented with 1-17-88
 ~ DUCLARATIONS FOR SCHOOL MEADER
  VARIABLE seronama (
      foundin "ASANTEADR ASSESSMENT . Dame"
    type STRING
formet 4-30s
 ) emperate MORAG
      NUM_rows 1
      awa_columns 50
      variable assename, 0 , 18
leader "Assessment name: "
      leader
     pickable 200
 DATOM assessmel (
      PART LOAM
      awa_columns 75
      variable assessme, 0 , 30
leader "Name of current assessment: "
                      200
 VARIABLE comment {
   foundin "ASANTEADR ASSESSMENT.desc"
     type STRING
format 9-66s
DARTM comment {
      aum columns 76
     variable comment, 0 , 10 leader "Comment: "
     leader
                   300
     pickable
VARIABLE Curnepasm (
type STRING
formet 4-30s
DATOM GERMAPAGE {
    RAME LOAD
    aum columns 77
    variable ournepass, 0 , 19
leader "Current map name: "
pickable mo
WIMDOW lin {
   aum_rows 1
aum_columns 78
line 0, 0
                 0, 0, 0,77
WINDOW REFRANCOS (
    num_rows 3
num_columns 78
```

```
comment, 1, 2
2, 0, 2,77
      1420
   WINDOW assumouni (
       num_columns
detum
                    78
       datus
                      comment, 1, 1
                    Gurmapaen, 2, 1
3, 0, 3,77
       dete

    Declarations for buttons used in majoraction footer

  TEXTLINE probetatet ( "REVIEW CORREST ASSESSMENT STATUS" )
  MUTTOW probetat (
      num columns 35
      textline probetatut, 0, 2
helpfile help/probetat.hlp
  TEXTLINE probetatizi2 ( "REVIEW ASSESSMENT STATUS" )
  BUTTON probetat2 {
      AUM_rows 1
      textline probstatut2, 0, 2
helpfile help/revstat.hlp
 TEXTLINE probdef ("ADD INFORMATION TO CURRENT ASSESSMENT" )
 MUTTON probdef (
     FEETLOWS
      num_columns 39
      taxtline probdef, 0, 2
helpfile help/probdef.hlp
     helpfile
 TEXTLINE probdef1 ("ADD TO ASSESSMENT DEFINITION" )
 BUTTON probdef1 {
     num_columns 31
    textline probdef1, 0, 2
helpfile help/probdef.hlp
TEXTLINE enalysis ( "AMALYEE DATA" )
BUTTON analysis (
     aum rows 1
aum columns 15
    textline analysis, 0, 2
helpfile help/dstenel.hlp
TEXTLINE reportses ( "MAKE A REPORT" )
BUTTOW reportgen {
    num columns 16
    textline reportgem, 0, 2
    helpfile help/reportgem.hlp
TEXTLINE VMchblistat ( "VIEW CHECKLIST FOR CURRENT ASSESSMENT" )
MOTTON weeklist (
   aum_rows 1
   num_columns 42
textline vwohklistrt, 0, 2
helpfile help/viewlist.hlp
        Declarations for majoraction footers
```

```
WINDOW majorastica (
                                              -header for problem status screen
         aum_columns 78
         1120
                            0, 0, 0,77
         title
                            "Alternative actions you can now take:"
                            problef, 2, 2, "NEW SCHIEN problefscreen"
enalysis, 2,56, "NEW SCHIEN viewchecklistscre
vechilist, 3, 2, "NEW SCHIEN viewchecklistscre
reportgen, 3,56, "CALL dummy2"
         button
        button
         betton
        button
   WINDOW majorastical ( "header for problem definition screen
        RUE POWE
        SWE_columns 78
                         0, 0, 0,77
         title
                          "Alternative actions you can now take:"
        betton
                     naclysis, 2, 2, "NEW SCREEN analysiscreen"
probetat2, 2,48, "CALL peprobet"
venklist, 3, 2, "NEW SCREEN viewchecklistscreen"
reportgen, 3,48, "CALL dumny2"
        bettos
        bettos
        button
  WINDOW majorastical (
                                           -header for data analysis screen
        BUN TOWN
        num_columns 78
        1120
                        0, 0, 0,77
        title
                          "Alternative actions you can now take:"
                    probdeft, 2, 2, "HEW SCREEN probdefscreen"
reportgen, 2,47, "CALL dumy?"
vwahklist, 3, 2, "HEW SCREEN viewchecklistscreen"
probstmt2, 3,47, "CALL peprobst"
        button
        button
        button
        bettoe
    }
- WINDOW majoraction3 ( whender for report generation screen
       BUR FORM
        NUM COLUMN 78
        محنة
                      0, 0, 0,77
                    "Alterative sctions you can now take:"

washlist, 2, 2, "MEW SCREEN viewchecklistscreen"

probdett, 2, 47, "CALL psprobst"

probdefi, 3, 2, "MEW_SCREEN probdefscreen"

analysis, 3,47, "MEW_SCREEN snalysiscreen"
        title
       betton
       betton
       button
 WINDOW majorastica4 (
                                       -header for view checklist screen
       he rows
       sum_columns 70
       معدد
                       0, 0, 0,77
       title
                        "Alternative actions you can now take:"
                     Probetat2, 2, 2, "CALL peprobet"
reportque, 2,47, "CALL dummy2"
probdef1, 3, 2, "NEW_SCREEN probdefecreen"
analysis, 3,47, "NEW_SCREEN analysiscreen"
       button
       button
       button
   }
 TEXTLINE loadglobal ("LOAD LOCAL DATA FOR GENERAL ACCESS")
 BOTTOM loadglobal (
     Bills_rows
     num_columns 39
textline loadglobel, 0, 2
helpfile help/loadglob.hlp
WINDOW majorastica5 (
                                         "footer for detabase housekeeping screen
      NOT TOWN
      NUM COLUMN 78
      line
title
                    0, 0, 0,77
                   "Alternative actions you can now take:"
loadglobal, 2, 2, "CALL dummy"
assessment, 3, 2, "CALL peprobet"
      button
SCREEN Stratey (
    title "MCR DATA EMTRY"
window mtrancom, 3,1
window mtratentry, 7,1
window majoraction3, 18,1
border YES
```

```
Declarations for problem status screen
  VARIABLE startdete {
      foundin "VARCHAR startdate.arr"
type STRING
foundt 4-21s
 DATOM startdate (
      awa_rows 1
       men columns 40
     variable startdate, 0 ,15 leader "Date started: "Data started: "
 VARIABLE lestdate (
       foundia "VARCEAR lastdate.arr"
      type STRING
format 4-21s
 DATOM lastdate {
     num_rows 1
num_columns 55
     variable lastdate, 0 , 28
lasdar "Data of last modification: "
pickable 80
 VARIABLE plantlast (
      foundin "VARCHAR plantlet.arr"
      type STRING
format 4-30s
 DATOM plenriest (
    ava columns 55
     variable plantlast, 0,19
leader "Last modified by: "
pickable 80
WIMDOW probetst (
     FAR LOAM
      atm_columns 78
     detum startdate, 0, 2 detum lastdate, 2, 2 detum planrlast, 4, 2
TEXTLINE objectit ("Work on a different assessment")
MUTTON abgass (
                                           -1-16-88

        num_cows
        1
        -1-16-88

        num_columns
        70
        -1-16-88

        taxtine
        chgasstxt
        0, 2
        -1-16-88

        helpfile
        help/probdef.hip
        -1-16-88

WINDOW chansess (
                                       -change to different assessment
     aun_columns 78
     button obgass, 0, 2, "CALL peologues"
         Declarations for MAVINGERIAL ASSESSMENT DEFINITION SCHOOL
TRATTLINE modmostxt ("Work with MOA information (number or type of minereft, missions, etc.)")
MUTTON modmon (
     aum rows 1
     aum_columns
                       76
     textline modeostat, 0 ,2
```

```
helpfile help/moswork.hlp
 TEXTLINE modelrint ("Work with MER information (number or type of mirroraft, missions, etc.)")
 NOTTON modetr (
      EME_TOWN
      aum columns 76
     textline modetrixt, 0 ,2
helpfile help/mirwork.hlp
 TRIFFLIRE modmaptxt ("Nork with map information (designate land uses, update maps)")
 NOTICE modesp (
     ME TOWN
     aum_columns 76
     textline modmeptxt, 0 ,2
helpfile help/mepwork.hlp
 THITLINE selection ("Actions you can now take to mid information to this assessment:")
 WINDOW selection (
     AVE POWS 7
      aum_columns 77
     textline selectat, 0, 1
                 modmap, 2, 1, "CALL dummy2"
modmar, 4, 1, "CALL pentrent"
modmon, 6, 1, "CALL dummy"
     button
     button
     betton
SCREEN 'probdefscreen {
    title "ENVIRONMENTAL ASSESSMENT DEFINITION"
    window assessment, 3,1
    window selection, 7, 2
    window anjorantion1, 18, 1
          Declaration for entering coordinates window
TEXTLINE estupplf
  ("Exter upper-left corner coordinates of area of current interest")
TRUTLING estlowet
  ("Enter lower right corner coordinates of area of current interest")
DATOM shwlat2 (
      Run_rows 1
      aum columns
                      25
     variable shwlat, 0 ,11
leader "Latitude: "
helpfile help/combmap.hlp
DATOM shwlong2 {
     RUM rows 1
RUM columns 25
      variable shwlong, 0 ,12
     leader
                  "Longitude: "
     helpfile help/combmap.hlp
WINDOW entocor (
    MOT_EOM
                    10
    am_columns
                  78
    textline
                   estupplf, 2, 2
    detun
                  entlet, 3, 1,
entlong, 3, 38,
                                         "CALL lat2ded Sent", "MERVALS"
    detun
                                        "CALL lon2ded feet", "MENVALS"
    textline
                  satlowit, 5, 2
shwlat2, 6, 1, "CALL lat2ded &show", "MENVALS"
    detus
    detus
                   shwlong2, 6, 38, "CALL lon2ded &show", "HEWVALS"
        Declarations for DATA AMALYSIS SCREEN
```

TRITLINE geodetinquit ("Make geodeta inquiries on map sureen")

```
SUFFOR geodeting {
     num rows 1
      num columns 76
     textline geodetingtxt, 0 ,2
helpfile help/geoing.hlp
 TEXTLINE companisafixt ("Compare zoise affects") -1-17-88
 BOTTON companies (
                                               -1-17-00
     ama rows 1
     num_columns 76
                                               -1-17-00
     taxtline compacisaftxt, 0 ,2
helpfile help/compacis.hlp
                                             -1-17-88
                                                    -2-06-88
                                    -1-17-08
 TENTIONE celemoiseftxt ("Calculate noise effects in specified area")
 NOTTON calcacted (
     BUB_POWS 1
     num_columns 76
     textline celcaciseftxt, 0 ,2
helpfile help/effcelc.hlp
 TEXTLES calcacisemptat ("Calculate noise emposure in specified area")
 MOTTON calcactsesp (
     num_rows 1
     textline calconousexptxt, 0 ,2
helpfile help/expcalc.hlp
TEXTLINE calcolitat ("Calculate quinklook (point) esposure estimate")
 BUTTON calcollook (
                1
     BWE FOW
     num columns 50
     textline calcoptlocktxt, 0 ,2
helpfile help/quklock.hlp
TEXTLINE estocortat ("Ester coordinates from keyboard")
 MUTTON entooor (
     REE FORM
     aum_columns 40
    textline entocortxt, 0 ,2 helpfile help/kbdmtry.hlp
TEXTLINE usemaptat ("Use map screen")
BUTTON usamap (
   hum_rows 1
    num_columns 30 textline
                                 weenaptat, 0 ,2
    helpfile help/usemap.hlp
TEXTLES selectasetzt ("Actions you can now take to analyze environmental assessment data:")
TEXTLES defgeoerestst ("Specify a geographic area of interest:")
WINDOW detamastion (
    FMD_2088
    aum columns 78
    textline
                   selectemetzt, 0, 2
                   onlogilook, 2, 2, "CALL dumny" onlogicatop, 3, 2, "CALL dumny"
    buttom
    buttom
    button
                   calcacisef,
                                   4, 2, "CALL domy"
                   celcacisef,
    button
                                    5, 2, "CALL dummy"
    button
                   geodating,
                                    6, 2, "CALL dumy"
    textline
                   defgeoerestat, 8, 2
                   usemap, 9, 2, "CALL dummy2"
entocor, 10,2, "ADD_WINDOW entocor 5 1"
   button
   betton
SCREEN analysiscreen {
   title "DATA AMALYSIS"
window assumnces, 2,1
window detanaction, 6,1
```

```
window majoraction2, 18 , 1
    border TES
 THISTLINE showmoretxt ("Show more assessments (if any)")
        shownore {
      aum_rows 1
      awn_columns
                    37
      textline shownoretxt, 0 ,2
      helpfile help/getassmet.hlp
 TRITLINE recells stat ("Recell one of the following assessments:")
 TENTLINE stressetzt ("Start a new assessment")
 MUTTON stanuars (
      FAMT LOAD
      awn_oolune
                    28
      textline
                     stnewsstat, 0 ,2
      helpfile
                    help/newsemat.hlp
 VARIABLE newspan {
foundin "VARCHAR n2bv.arr"
      type
                 STRING
      format
                 4-30#
DATCH SOURSES (
     PAR LONS
     Name of the same
                     40
                     BOWLELEN,
      veriable
                                 0,7
                     "Nume : "
     Leader
     helpfile
                     help/stnewnss.hlp
WINDOW BO
               • {
                     40
WINDOW stanmes {
                    78
                     Stnewnss, 0, 2, "ADD_WINDOW new
                              "UPDATE DATCH newspan",
"CALL penwasen n2bv.arr"
                          sore, 0, 41, "CALL abunch"
WENDOW recalass {
     NEW YORK
                    1
     Sum_columns
                    76
     testilse
                    recallsestrt, 0, 2
WINDOW Gurassbut {
      BUR TOWN
                     10
                   2
       aum_columns
       button malbut00,
                          0, 0,
                                     "CALL ASAMoona deplemit[0].arr"
                                     "CALL ASAMoonn deplanit[1].arr"
"CALL ASAMoonn deplanit[2].arr"
       button
              mulbut01, 1, 0,
       button
               malbut02, 2, 0,
       button
               mulbut03, 3, 0,
                                     "CALL ASAMoona depimult[3].arr"
       button
               mulbut04,
                           4, 0,
                                     "CALL ASAMoona deplault [4] .arr"
       button
               mulbut.05,
                           5, 0,
                                     "CALL ASAWoonn deplanit[5].arr"
               mulbut06, 6, 0,
mulbut07, 7, 0,
       buttom
                                     "CALL ASAMoona deplault[6].arr"
       button
                                     "CALL ASAMoonn depletat [7] .err"
      button
               mulbetos, s, o,
                                     "CALL ASAMoonn deplacation .arr"
                                     "CALL ASAMoonn deplement[9].arr"
      button
               mulbut09,
         Gurassbut1 {
      aum_rows 10
      button malbut10, 0, 0,
                                     "CALL ASAMoonn depleuit[10].arr"
      button mulbut11, 1, 0,
                                    "CALL ASAMoonn deplault[11].arr"
      button mulbut12, 2, 0,
                                    "CALL ASAMoonn deplanit[12].arr"
      button mulbut13, 3, 0,
button mulbut14, 4, 0,
                                    "CALL ASAMoonn deplault[13].arr"
                                    "CALL ASAMoons deplacatt[14].err"
```

```
button malbut15, 5, 0,
button malbut16, 6, 0,
button malbut17, 7, 0,
button malbut18, 8, 0,
button malbut18, 9, 0,
                                                "CALL ASAMoonn deplanit[15].arr"
                                                "CALL ASAMocan deplant [16] .arr"
                                                "CALL ASAMoonn deplant [17] .arr"
                                                "CALL ASAMoonn deplacate[18] .arr"
                                                "CALL ASAHoona deplanit[19].arr"
     TEXTLINE retuces ("Continue without selecting assessment")
     BUTTOW retessent (
        REN_TODA
        atta columns 40
        testline
                       retmonss, 0 ,2
       helpfile help/monssess.hlp
    WINDOW retwoes (
       FAST LOAD
       2001000 76
       betton
                    retassent, 0,1, "CALL misto", "CALL peprobet"
    SCHEEK characterscreek (
         title
                     "SELECT ANOTHER ASSESSMENT"
         Window
                      88624EGCE, 2,1
         window
                      Strownes, 6,1
         window
                      receless, 8,1
         window
                      Gurasabut, 9,3
         wiadow
                      Gurassbutl, 9,42
         window
                      Gurdet, 9,5
Gurdet1, 9,44
Fetwones, 20,2
         window
         window
         border
                      774
   " Declarations for NEW ASSESSMENT DEFINITION
   VARIABLE entdessi {
       foundin "VARCHARGO entdeed[0].arr"
type STRING
formet 0-60s
  DATTM estdeed1 (
       AUM_rows 1
AUM_columns 75
variable entdesd, 0 ,0
helpfile help/entdesd.hlp
 VARIABLE estdess2 {
       foundin "VARCHAR60 entdesd[1].arr"
      type
format
                   STRING
                 4-60s
DATCH entdessi (
      ham rows 1
ham columns 75
variable entdesst, 0,0
halpfile halp/entdess.hlp
VARIABLE entdess3 {
     Foundin "VARCHAR60 estdess[2].arr"
type #FRIES
foundt 4-60s
  3
DATOM estderes {
   num_rows 1
num_columns 75
    variable entdered, 0 ,0 helpfile help/entdesc.hlp
VARIABLE entdesc4 (
    foundin "VARCHARGO entdeed[3].arr"
     type
                 STRING.
                 4-60g
```

```
DATOM estdeed (
       num_rows 1
       aum_columns 75
      variable estdesc4, 0 ,0
helpfile help/estdesc.hlp
 TEXTLIRE entdescript ("Please enter a brief description for this assessment")
  newssman (
ann rows 1
ann columns 77
variable newssman, 0 ,35
leader "Enne of new assessment definition:"
pickable NO
)
 DATTM nowassam (
 WINDOW memandass {
      num_rows $
                   noveconum. 1.3
       textline entdeedtat, 3,4
       detum entdesc1, 4,4
detum entdesc2, 5,4
detum entdesc2, 6,4
detum entdesc4, 7,4
horder YES
                    TES
        Declarations for database housekeeping screen
 TENTLINE updateinfotzt ("Update information")
 BOTTOM updateinfo (
      aum_rows 1
aum_columns 50
      textline updateinfotxt, 0 ,2
helpfile help/updatein.hlp
 TEXTLER asstabletzt2 ("Frint list of all columns in an assessment's tables")
 NUTTOW asstable2 (
      aum_rows 1
aum_columns 55
      taxtline asstabletxt2, 0 ,2
helpfile help/asstable.hlp
 TRITLINE asstabletntl ("Print list of an assessment's tables")
NOTICE asstable1 (
     ava_rows 1
ava_columns 50
      textline asstabletxt1, 0 ,2
helpfile help/asstable.hlp
TRITLINE assesstat ("Frint list of all assessments")
MUTTON mesons (
     aum_rows 1
      ava columns 50
     textline assesstat, 0 ,2
helpfile help/sohelp.hlp
TENTEDES didetetat ("Frint all detabase dates")
NOTION didate (
     hum_rows 1
hum_columns 50
     textline dbdstetxt, 0 ,2
helpfile help/dbdste.hlp
TEXTLINE dbhsakpgtxt ("RARNING: Actions you take on this screen affect ASNE'S permanent databases!")
```

```
dibbsekpgestice
                            13
             Bun ro
             aum_oolu
                            78
            1120
                            0,0, 0,77
            textline
                            dbheelpgtzt, 2,2
                            dodate, 4,2, "CALL dummy" assess, 6,2, "CALL pratabo"
            bettos
            button
                            .....
                            asstable1, 8,2, "CALL SUPPOPT 1"
asstable2, 10,2, "CALL SUPPOPT 2"
            button
            button
                            updateinfo, 12,2, "CALL dumny
 SCREEN dhiseippscreen (
                   "DATABASE BOUSEKEEPING"
       title
       window
                   dbhsekpgaction, 2,1
                   majorastica5, 18,1
                   TES
 WINDOW quickessbut {
                       10
        Bun_rows
                      2
         aum columns
        button mulbut00, 0, 0,
                                         "CALL SOprist deplault[0].arr"
        button mulbut01, 1, 0,
                                         "CALL SUprist deplemit[1].arr"
         betton
                 mulbut02, 2, 0,
                                          "CALL SOprint deplanit[2].arr"
         betton
                 mulbut03, 3, 0,
                                          "CALL SUprint depimult[3].arr"
                                          "CALL SUprist deplault[4].err"
         button
                  mulbut04,
                               4, 0,
         betton
                  mulbut05,
                                         "CALL SUprist deplacit[5].arr"
                  mulbut06,
        button
                                         "CALL SUprint deplacat[6].arr"
        button
                  mulbut07,
                               7, 0,
                                         "CALL SOprist deplacate[7].err"
        bettom mulbutos, 8, 0,
                                         "CALL SUprist depleatt[8].arr"
        button
                  mm1but09. 9. 0.
                                         "CALL SUprist deplealt[9].arr"
WENDOW quickessbut1 {
        num rows 10
num columns 2
        Par Lone
        button mulbut10, 0, 0,
button mulbut11, 1, 0,
                                         "CALL SUprint deplault[10].arz"
                                         "CALL SUprint deplault[11].arr"
        button mulbut12, 2, 0,
                                         "CALL SUprint deplault[12] .arr"
        button mulbuti3, 3, 0,
                                         "CALL SUprist deplacet[13].arr"
        button
                  mulbut14, 4, 0,
                                         "CALL SOprist deplault[14].err"
        button mulbut15, 5, 0,
                                         "CALL SUprist deplault[15].arr"
        betton mulbutle, 6, 0, betton mulbutle, 1, 0, betton mulbutle, 8, 0, betton mulbutle, 9, 0,
                                         "CALL SUprist deplacit[16].arr"
                                         "CALL SUprist deplacit[17].arr"
                                         "CALL SUprist deplsalt[18].arr"
                                         "CALL SUprist deplett[19].arr"
TEXTLINE susalect ("Print list of SUPERUSER's tables")
TEXTLINE hqualect ("Print list of READQUARTERS' tables")
BOTTON susalest (
     num_rows 1
num_columns 50
      textline suselect, 0 ,2
helpfile help/nohelp.hlp
BUTTON hquelect (
     num_rows 1
num_columns 50
     tartline hquelect, 0 ,2
helpfile help/nohelp.hlp
WINDOW retwopen (
   BUR_POWE
   aum columns 76
                 retassent, 0,1, "CALL ulista", "NEW_SCREEN dbheekpgsureen"
WINDOW hqsusalect {
   BWB_FOWS
   aum_columns 76
                hquelect, 0,1, "CALL SUpropt 3" suselect, 2,1, "CALL SUpropt 4"
   button
   bettoe
SCREEN slossareen (
```

```
title "SKLECT ASSESSMENT FOR DATABASE PRINTOUT"
window hquuselect, 4,1
window recalass, 8,1
window quickassbut, 9,2
window quickassbut1, 9,42
window quickassbut1, 9,42
window quickassbut1, 9,44
window curdet1, 9,44
window retwoprn, 20,2
horder TES
}
```

```
Declarations for view CHRCK-LIST SCHOOL
   TEXTLES otherchilst ("View another checklist")
  BUTTON otherchilst (
                    num rows 1
num columns 25
textiine otherchkist, 0, 2
helpfile help/othrchk.hlp
  TEXTELOGE fone12 {
                     filenme trtblk/fonsi2.trt
sum_cos 10
sum_columns 76
border TES
 WINDOW fondi2 (
                      num_columns
 Filename to a management of the search of th
                                                                                     tstblk/fossil.tst
                                                                 teti
10
16
76
725
                       245_001
WINDOW foasil {
   aum_rows 13
   aum_dolumns 78
   taxtbloak foasil, 0,1
   buttoa otherchkist, 12,2, "REMOVE_WINDOW"
   TEXTRICCE oster3 {
                  filename tati
num_rows 10
num_columns 76
                                                                                     tatblk/cetex3.tst
                      border
                                                                                   YES
 WINDOW cetar3 {
                    W onte-
htm rows 13
atm_columns 78
textblock ontex3, 0,1
otherchkist, 12,2, "REMOVE_WINDOW"
TEXTELOGE ostax2 {
                  fileneme
                                                                                   txtblk/catex2.txt
                     aum_rows
aum_columns
border
                                                                                   10
                                                                          10
76
                                                                                   THE
WINDOW cates2 {
```

```
nun_oolunas
         tastbloak catar2, 0,1
tastbloak otheraklet, 12,2, "NOMEOVE NINDON"
   240 2000
         AWA COLUMN
                         76
         border
                           124
   WINDOW outer1
                        (
        BANT LONG
                            13
         namicolumns
                        78
         textblock
                            cetezi, 0,1
                         otherchklst, 12,2, "REMOVE WINDOW"
         button
  THISTLES foneithm ( "Show documentation necessary for finding of no significant impact")
  BOTTOM fonsi2 {
       NUM TOWN 1
        SWE COLUMNS 72
        textline fonsi2txt, 0, 2
helpfile help/fonsi2.hlp
  TROTLING foasiltst {
"Show MEPA bases for finding of no significant impact (FORSI)"}
  BUTTON fondil (
        AUM_POWS 1
AUM_columns 65
        tertline fonsiltxt, 0, 2
helpfile help/fonsil.hlp
  TEXTLINE catesitat ("Show documentation needed for categorical exclusions")
 SUPPLIES SECTION
       aum_rows 1
aum_columns 55
       testline cetaritat, 0, 2
helpfile help/cetari.hlp
 THITLING osterity (
"Show examples of proposed actions qualifying for categorical exclusions")
 BUTTON catazi (
      num_rows 1
      hum columns 74
textline cetax2txt, 0, 2
helpfile help/cetax2.hlp
TEXTLINE optoritet ("Show NEPA bases for categorical analysicas (CATEX)")
MOTTON OSTABL (
      RUE_rows 1
      num rows 1
num columns 55
taxtline cateritat, 0, 2
helpfile help/cateri.hlp
WINDOW vechblist {
    Rum rows 11
    Rum columns 78
    line 0,0,0
        aum_columns 78
lias 0,0,0,0,77
buttos ostari,
buttos ostari,
buttos fossii,
buttos fossii,
                                      2,2, "ADD_WIMDOW cater1 3 1"
                                     4,2, "ADD WINDOW Catar2 3 1"
6,2, "ADD WINDOW Catar3 3 1"
8,2, "ADD WINDOW Consil 3 1"
                                       10,2,
                                                    "ADD_WINDOW foom12 3 1"
SCREEN viewshecklistscreen (
    title "VIEW CEECELIST"
              weaklist, 2,1
majoraction4, 18,1
     window
     Window
```

```
border TES
   SCHEME probetatsurees {
   title "ENVIRONMENTAL ASSESSMENT STATUS"
      mainscreen YES
      border
                  124
                  assamocm, 3,1
probstat, 7,1
chgassess, 16,1
majoraction, 18,1
      window
      window
      window
      window
   -MED OF MANY SOF
  -INCLUDE STATEMENT FOR THE ASAN typedof DEFINITIONS NUMBER TO
COMPILE USUAS.C WITHOUT INCURRING THE WHATH OF THE COMPILER
  DICTOR ASAN.E
  - DECLARATIONS IN NON-LEXICAL OBJECT ENGUMENT
  - Multiple Choice Space
  VARIABLE muldatoo (
     foundin "VARCEARIO deplanit[0].err"
     type STRING
format 4-30s
 DATOM muldatoo (
     RUM_FOWS 1
    NUM_columns 32
Variable muldatoo, 0, 2
pickable NO
 VARIABLE muldet01 (
    foundin "VARCEAREO deplement[1].arr"
    type STRING
format 4-30s
 DATOM muldat01 {
    Bun_rows 1
    aum_columns 32
    variable muldat01, 0, 2
pickable NO
 VARIABLE muldet02 {
    foundin "VARCEARSO deplault[2].arr"
    type STRING
formet 4-30s
DATOM muldet02 (
    RUM_FOWN 2
    Num columns 32
   variable muldat02, 0, 2
pickable 20
VARIABLE muldat03 (
   foundin "VARCHAR30 deplanit[3] .arr"
   type STRING
formet 4-30s
DATOM muldatos (
```

```
am_rows 1
      num columns 32
variable muldat03, 0, 2
pickable 20
  VARIABLE muldet04 {
foundin 'VARCHAR30 deplault[4].arr'
type STRING
foundt 4-30s
  DAFOM suldet04 (
num_rows 1
num_columns 32
     variable muldet04, 0, 2
pickable MO
  VARIABLE muldet05 (
     Toundia "VARCHARJO deplmuit[5].arr"
type STRING
format 4-30s
  DATOM muldat05 {
     htm_rows 1
     aum columns 32
     variable muldet05, 0, 2
pickable NO
  VARIABLE muldetos (
     foundin "VARCHAR30 deplault[6].arr"
     type STRING
formet 4-30s
 DATOM muldet06 {
     RUM rows 1
RUM columns 32
     variable muldat06, 0, 2
pickable 20
 VARIABLE muldet07 {
    foundin "VARCHAR30 deplanit[7].arr*
type string
format %-30s
 DASTM muldato7 (
    Num rows 1
Num rolumns 22
variable muldet07, 0, 2
pickable 30
 VARIABLE muldatos (
    foundin "VARCHAR30 deplault[8].arr"
    type STRING
format 4-30s
DATOM muldatos (
    AME_TOWN 1
    Aum_columns 32
    variable muldet08, 0, 2
pickable 20
VARIABLE muldatos {
   foundin "VARCHAR30 deplanit[9].arr"
type STRING
format 4-30s
 }
DATOM muldatos (
  AWA_TOWN 1
   aum_columns 32
   variable muldet09, 0, 2
pickable MO
```

```
VARIABLE maldet10 {
    foundin "VARCHARSO deplanit [10] .arr"
    type STRING
format 4-30s
 DAFOM muldet10 (
    hum_rows 1
    num_columns 32
    variable muldet10, 0, 2
pickable 20
 VARIABLE muldet11 {
    foundin "VARCHARIO deplement[11] .arr"
    type STRING
format 4-30s
 DATOM muldetii (
    num_rows 1
num_columns 32
    variable muldet11, 0, 2
piskable 20
 VARIABLE muldet12 {
    foundin "VARCEARIO deplault[12].arz"
    type STRING
formet 4-30s
 DATOM muldet12 {
    Bum_rows 1
    BWE_columns 32
    variable muldat12, 0, 2
pickable NO
 VARIABLE muldet13 {
    foundia "VARCHAR30 deplanit[13].arr"
type STRING
formet 4-30s
DATOM muldet13 (
   aum rows 1
aum columns 32
   variable muldet13, 0, 2
pickable 20
VARIABLE muldet14 (
   foundin "VARCHAR30 deplault[14].arr"
   type STRING
formet 4-30s
DATOM muldet14 (
   RUB_rows 1
RUB_columns 32
   variable muldet14, 0, 2
pickable NO
VARIABLE muldet15 (
   foundin "VARCEARSO deplesat [15] .arr"
   type STRING
formet 4-30s
DATOM muldet15 (
   num_rows 1
   num columns 32
   Variable muldet15, 0, 2
pickable 20
VARIABLE muldet16 {
  foundin "VARCHAR30 deplanit[16].arr"
```

```
type STRING
formet 4-30s
  DATOM muldet16 {
     aum_rows 1
      aum_columns 32
     variable muldet16, 0, 2
pickable NO
  VARIABLE muldet17 (
     foundia "VARCEAR30 deplault[17].arr"
     type STRING
formet 4-30s
 DAFUM muldet17 {
     num_rows 1
num_columns 32
     variable muldet17, 0, 2
pickable MO
 VARIABLE muldet18 {
     foundia "VARCHARIO deplesatt[18].arr"
     type STRING
formet 4-30s
 DATOM muldet18 (
     hum rows 1
hum columns 32
     variable maldet18, 0, 2
pickable 20
 VARIABLE muldet19 {
foundin "VARCHAR30 deplault[19].arr"
type STRING
format 4-30s
 DATOM muldet19 (
    aum rows 1
aum columns 32
    variable muldat19, 0, 2
pickable 20
 ~ Planner attributes
 VARIABLE plearnes (
    foundin "VARCHAR plearmen.arr"
type STRING
formet 4-30s
DATOM plearnes (
   Num rows 1
    aum_columns 50
   variable planram, 0, 20
leader "Your name, please:"
helpfile help/username.hlp
VARIABLE password (
  type STRING
formet 4-10s
DATOM password {
   RUM_rows 1
   aum_columns 55
   variable password, 0 , 23
leader "Flease enter password:"
helpfile help/password.hlp
- Other General Storage for Varification, Tests, Etc.
```

.

```
VARIABLE sountres {
foundis "VARCHAR sibv.arr"
         type
                           FERING
        formet
                         4-30s
  - Noise Source Attributes
  VARIABLE sraid {
      foundin "VARCHAR stuid.art"
      type symmet
 DAFOM miram (
num_rows 1
num_columns 60
     variable sreid, 0, 21
leader "Hame of current MER:"
pickable 20
  VARIABLE srodess (
     foundin "VARCEAR STONESS.AFT"
type STRING
formet 4-54s
 DATOM spods (
         Name Town
         aum_columns 76
         Variable srodesc, 0, 22
leader "Description:
                     help/mtrdesc.hlp
         helpfile
 DATOM strdess (
        Par Lone
         ata columns 70
         variable srodesc, 0, 7
leader "Note: "
pickable NO
         leader
         pickable
 VARIABLE schedule (
    foundin "VARCHAR srosched.arr"
type STAIRG
formet 4-50s
DATOM schedule (
       hum rows 1
aum_columns 73
variable schedule, 0, 22
leader "Scheduling activity: "
helpfile help/schdmtr.hlp
VARIABLE stoorig {
foundin "VARCHAR stoorig.arr"
type STRING
formet 4-50s
DATCH origits (
   ava_rows 1
ava_columns 73
   variable smoorig, 0, 22
leader "Originating activity:"
helpfile help/origatr.hlp
~ MTR Attributes
```

```
VARIABLE Gurartos (
                                                           - CURRENT Set
      foundin "VARCEAR Gurartos.arr"
      type STRING
format 43s
  DATOM curertoc (
         PART LOM
          ave columns
                         27
          veriable
                          curartos, 0,16
          leader
                          "ARTCC:
          belpfile
                       help/ourartos.hlp
  VARIABLE curvidright (
type INTEGER
      formet $2d
  DATOM ourwidnight (
         aum rows 1
aum columns 25
        FEET LONG
         variable curvidright, 0,16
leader Width (right): *
helpfile help/curvidrt.hlp
 VARIABLE curvidleft (
type INTROER
     formet 42d
 DATOM ourwidleft (
       num rows 1
num columns 25
variable curvidleft, 0,16
lander "Width (left): "
        helpfile help/ourwidlf.hlp
 VARIABLE cerhighelt (
    foundin "ALTSPEC curhighalt.spec"
     type STRI
formet 49s
                  STRING
 DATOM carhighalt (
      am rows 1
       aum_columns 27
        variable ourhighalt, 0,16
leader "Eigh altitude: "
helpfile help/ourhialt.hlp
VARIABLE curlowalt {
foundia "ALTEREC curlowalt.spec"
type STRING
formet $96
VARIABLE preartod (
foundia "VARCHAR preartod.arr"
type #TRING
formet $3s
                                                      - PREVIOUS Set
   DATOM presentes {
      AND YOUR 1
AND COLUMNS 8
Veriable presented, 0,4
pickable 80
VARIABLE prewidingst (
   formet $2d
DATOM prewidright {
      NUM rows 1
       variable previdright, 0,4
```

```
pickable No
  VARIABLE providingt (
     type INTEGER formet 42d
   DATOM providleft (
        am row 1
         Variable provideft, 0,4
pickable 20
 VARIABLE prehighelt (
foundin "ALTEFEC prehighelt.spec"
type STRIBU
format 49s
 DAFOM prehighelt (
        ava_columns 10
        variable prehighelt, 0,1
pickable 20
 VARIABLE prelomatt (
      foundin "ALTEFEC prelounit.spec"
type STRING
formet the
 }
 - Coordinates
 VARIABLE entlat (
foundin "COORDINATE ent.let"
type STRING
formet %-13s
                                                      - CURRENT OF BRIER Set.
 VARIABLE entlong {
   foundin "COORDINATE ent.lon"
   type STRING
      type smint
format 4-13s
VARIABLE shwist {
   foundin 'COORDINATE show.lst'
   type STRING
                                                   ~ PREVIOUS or SHOW Set.
      type
format
VARIABLE shwloag {
foundin "COORDINATE show.loa"
type STRING
                4-130
- Mavigation Point Properties
VARIABLE ourflistyp (
                                                   - CORREST OF BETER Set
     foundia "VARCHAR ourfirtyp.arr"
type STRING
foundt $12s
VARIABLE ourfindist (
     type IFFEGER
VARIABLE ourflured {
               IFFROR.
     type
      format
VARIABLE ourfixed (
     foundin "VARCEAR Gerfixed.err"
```

```
type
format
                               STRING
                              45e
    }
     VARIABLE cursevpt (
          foundin "VARCEAR Gurnavpt.arr"
                        #29.3363
$3.e
           format
    VARIABLE prefixtyp (
foundia "VARCHAR prefixtyp.arr"
type STRING
foundt 912s
                                                                 - PREVIOUS or DISPLAY Set
    VARIABLE profindist (
          type DFFMER
    VARIABLE prefixred (
type invader
formet $3d
   VARIABLE prefixed (
foundia "VARCHAR prefixed.arr"
type STRING
formet 45s
          type
format
                              450
   VARIABLE pressvpt {
    foundin "VARCEAR pressvpt.arr"
                      STRING
          type
format
   - Aircraft Parameters
   VARIABLE comme {
foundin "VARCEAR ac name.err"
type STRING
formet $-12s
  DATOM someme (
          num_rows 1
aum_columns 25
variable accume, 0, 10
leader "Aircraft:"
pickable NO
  VARIABLE accement (
     foundin "VARCHAR tid.arr"
type STRING
formet 4-12s
  DATOM acmemal (
          aum_rows 1
          aum_columns 35
         variable accessed, 0, 16
leader "Aircraft name:"
helpfile help/straircr.hlp
    }
  - Operations Specification
 VARIABLE day {
foundin "OFERATIONS ops[0].day"
type INTEGER
formet 44d
                                                               - CENTERIC OF SAMUARY
٠,
 VARIABLE junday (
foundin "OPERATIONS ope[0].day"
type INTEGER
formet +44d
```

```
VARIABLE jumite (
foundia "OFFRATIONS ope[0].mite"
type DITECT
foundt 04d
   VARIABLE febdey (
foundin "OFERATIONS ope[1].day"
type INTRGER
formet 44d
   VARIABLE febrito (
foundia "OFERATIONS ope[1].mite"
type DFFMUR
foundt %4d
    }
   VARIABLE marday (
foundin "OFFRATIONS ope(2).day"
type INTEGER
format 44d
   VARIABLE mermits {
   foundin "OFERATIONS ope[2].mits"
   type DEFECTA
   foundt 04d
  VARIABLE sprday {
   foundin "OPERATIONS ope[3].day"
   type INTEGER
   foundt #4d
  VARIABLE apraits (
foundin "OPERATIONS ope[3].mits"
type INTEGER
formet 04d
  }
  VARIABLE mayday (
foundin "OPERATIONS ope[4].day"
type INTRODER
formet 44d
 VARIABLE maynite {
   foundin "OPERATIONS ope[4].nite"
   type INTEGER
   formet 94d
 VARIABLE junday (
foundin "OPERATIONS ope[5].day"
type INTROSER
format +44
VARIABLE junnite (
foundin "OFERATIONS ope[5].nite"
type DEFECT
formet #44
VARIABLE julday (
foundin "OPERATIONS ops[6].day"
type INTEGER
format 44d
VARIABLE julnite (
foundin "OPERATIONS ops[6].nite"
```

```
type memora
format 44d
    VARIABLE augday (
foundia "OPERATIONS ope[7].day"
type REFEREN
formet #44d
    VARIABLE engaite (
foundin "OPERATIONS ope[7].mite"
type ZHINGER
formet 04d
   VARIABLE sopday {
    foundin "OPERATIONS ope[8].day"
    type INTEGER
    foundt +44d
   VARIABLE sepaite {
foundia "OFFENTIONS ope[8].mite"
type DFFENTA
formet 84d
  VARIABLE octnite {
foundix "OFERATIONS ope[9].mite"
type INTROCK
formet 04d
  VARIABLE movday (
foundin "OPERATIONS ope[10].day"
type INTRACE
format 04d
  VARIABLE moveite (
foundin "OFERAFIORS ope[10].mite"
type INFRARR
formet 644
 VARIABLE decday (
foundin "OFERATIONS ope [11].day"
type DETECT
format 444
 VARIABLE decnite {
   foundin "OPERATIONS ops[11].mite"
   type INTEGER
   formet #444
 - Missions
VARIABLE missens (
foundin "VARCHAR misslabl.arr"
     type STRING
formet 4-7s
DATOM missemel (
      hum rows 1
         variable missame, 0, 26
leader "Emme of current mission: "
pickable 20
VARIABLE nomisseme {
```

```
foundia
                           "VARCEAR GLd.arr"
          type
                           673.TB4
                           4-7s
 DASCH misses {
         BULL FORM
         ave columns 24
         veriable
                          mennisamo, 0, 14
                           "Mission name:"
                           help/missness.hlp
         helpfile
 TEXTLIES select {
     "Right now you can type ? for help, <CTRL> C to quit, or move the exreor")
 EMERITIES INTLOFED ("ALEM CENERAL INCLUMNATION VECOL SELE MOCHUM.)

EMERITIES POSSESS ("AMERICA DELINGUALING VECOL SELECTION.)

EMERITIES CONTROL VE EMAINMENTAIN VERSENMENT.)
 TENTUME selecuistri ("Select aircraft and mission for MER")
TENTUME modernmentrixt ("Modify Gurrent MER")
 TEXTLEM object: ("Select enother MER")
 TEXTLES next nevotat ("Enter next nevigation point")
 TENTLIES ourprotest
 ( "CURRENT
                  PREVIOUS
                                                         CONCERT
                                                                        PREVIOUS" }
 THEFLUI CUrpretati
 TEXTLINE cancelmittit ("Cancel this MIR data entry")
 TEXTLES seventrint ("Seve this MER")
 TEXTLINE showmorestrixt ("Show more MIR names (if any)")
TRITLUME recallstrirt ("Recall one of the following MERE:")
TRITLUME stnewartzt ("Start new MER")
TRITLUME definestrirt ("Enter route waypoints")
TEXTLINE actst ("Aircraft:")
TEXTLINE opensonist ("Operations are sessonal")
THISTLINE opensount: ("Operations are seasonal")
THISTLINE opthraystit ("Operations are even throughout year")
THISTLINE instrict:
THISTLINE instrict:
THISTLINE instrict:
THISTLINE canonistit:
THISTLINE canonistit:
THISTLINE sthemistit:
THISTLINE sthemistit:
THISTLINE sthemistit:
("Start new mission")
TEXTRLOCK introtat (
    hum_rows 14
hum_columns 76
     border YES
filename txtblk/intro.txt
BUTTON RESONANT
    nun_rows 1
nun_columns 38
   Man Lone
   textline envass, 0, 2
belpfile belp/assess.hlp
BOTTON chamtr (
   THE PORT
    awa_columns
                       35
    textline
                obgetrier, ...
belp/selstr.blp
                       obgestrick, 0, 2
   belpfile
MOTTON housekeeping (
   num rows 1
   textline houselp, 0, 2
helpfile help/houselp.hip
NOTICE introtat (
   BUR FOWS 1
   num_columns 56
   textline
                    imtrotat, 0, 2
                 istrouxe, u, _
belp/introbel.hlp
   helpfile
```

```
BUTTON modernmetr (
     amarons 1
     ava_columns 35
     textline modeumstrixt, 0, 2
helpfile help/mohelp.hlp
- mo help here our this button doesn't do snything
  WINDOW password (
      num_rows 1
num_columns 60
      detu
                    password, 0, 5
  TEXTLEME tititat
  ("Developed for Noise and Sonic Boom Impact Technology Program")
 TEXTLINE tit2tst ("under U.S. Air Force Contract F33615-86-C-0530")
 TEXTLES titStat ("by BBM Laboratories, Inc.")
 TEXTLEM titétat ("February, 1900")
 TEXTLINE titStat
 {"Unreleased demonstration of Prototype Version...Not for General Use"}
 TRITLING diafo ("Dose viewing general information on ASAR")
 BUTTOW diafo (
     MEN_TOWN
     aum_columns 76
     textline diafo, 0, 2
 WINDOW introduction (
     Par Lone
                 15
     num_columns 78
taxtblock introtxt,
                   introtat, 0, 1
diafo, 14, 1, "REMOVE_WINDOW"
 WINDOW introvindow1 (
     RWS_ZOWS
     RUM_FOWN 21
RUM_GOLUMNS 78
           1, 0, 1, 77
     title
                   "ASSESSMENT SYSTEM FOR AUXCRAFT HOUSE (AMAI)"
     textline
                   tititmt, 2, 9
     textline
                   tit2txt, 3, 16
     textline
                   titStst, 4, 27
                   tit4txt, 5, 32
tit5txt, 7, 6
8,0, 9,77
     textline
     textline
     detun
                   plearnen, 12, 1,
                    "CALL VCAPITAL Splanman"
                   "ADD_WINDOW password 15 3",
                   "UPDATE DATUM password",
                   "REMOVE WINDOW",
                   "CALL pwaheak"
 }
WINDOW introwindow2 {
    NW_TOWS
     ava_columns 78
    محند
                   5, 0, 5, 77
    textline
                   selast,
                                  6, 3
                  introtxt, 7,15, "AND WINDOW introduction 3 1" assessment, 8, 2, "CALL peprobet" housekeeping, 8,44, "CALL pedbhack"
                   introtat,
    button
    button
SCREEN firstsureen (
   titlescreen YES
   border
                 YES
                 "ASSESSMENT SYSTEM FOR AIRCRAFT MOISE (ASAM)"
   title
                 introvindow1, 1,1
introvindow2, 13,1
MUTTON openson (
   BUR FOWS 3
   aum_columns 41
              opsessortut, 1, 3
```

```
border YES
helpfile help/whlops.hlp
  NOTION opsteady {
     aum_columns 41
     tastline opthrwystrt, 1, 3
border THS
helpfile help/evenops.hlp
  BOTTON mulbetoo {
    aum_columns 1
helpfile help/mulbut.hlp
  NOTICE mulbet01 (
    RWA_rows 1
     num columns 1
     helpfile help/mulbut.hlp
  NOTTON malbato2 (
    ME TOW 1
    num_columns 1
    helpfile help/mulbet.hlp
 MOTTOW malbatos (
    AME_pows 1
    aum_columns 1
helpfile help/mulbut.hlp
 MOTTOW mulbut04 (
    sum columns 1
    helpfile help/mulbut.hlp
 BOTTOW malbut05 {
    num_rows 1
num_columns 1
helpfile help/nulbut.hlp
 MOTTON malbatos (
    BWE_columns 1
    helpfile help/mulbut.hlp
NOTICE mulbut07 {
   AUB_rows 1
    num_columns 1
   helpfile help/mulbut.hlp
BOTTON mulbetos (
   htm_rows 1
   num columns 1
   helpfile help/mulbut.hlp
BOTTON malbetos (
   RUM_rows 1
   aum columns 1
   helpfile help/mulbut.hlp
BOTTOW mulbet10 (
  RME_FORM 1
   NUM Columns 2
   helpfile help/mulbut.hlp
BOTTON malbatil (
  EME_rows 1
EME_rows 2
belpfile help/sulbut.hlp
```

```
1
   MOTTOW mulbut12 {
     RWE_rows 1
      aum_columns 2
      helpfile help/mulbut.hlp
  BUTTOW mulbet13 (
     num_rows 1
num_columns 2
     helpfile help/mulbut.hlp
  MOTTOW mulbut14 (
     NUM FOWS 1
      awa_columns 2
     helpfile help/mulbut.hlp
  MOTTON malbet15 (
     RMM_rows 1
     awa_columns 2
     helpfile help/mulbut.hlp
  BOTTON mulbet16 {
     NUM columns 2
     helpfile help/mulbut.hlp
 SOTTON mulbet17 (
num_rows 1
num_columns 2
     helpfile help/mulbut.hlp
 BOTTON mulbet10 {
    aum_columns 2
     helpfile help/milbut.hlp
 BOTTOW mulbut19 (
    RUM_FOWS 1
    awa_columns 2
    helpfile help/mulbut.hlp
          Declarations for MIR DATA ENTRY SCREEN
NOTICE stanuis (
    num columns 40
textline stneumistrt, 0,2
helpfile help/stneumis.hlp
NOTTON Selectio (
     PART LOAD
                      1
      am columns
                    40
     taxtline selamistxt, 0, 2
helpfile help/selam.hlp
YESTLINE spectrumistri ("Specify new mission")
NOTION speciments (
     ow spectaturis (
num_rows 1
num_columns 30
textline spectaturistri, 0, 2
helpfile help/neumiss.hlp
WINDOW mirdatentry (
  num_rows 8
num_columns 78
  betton object, 0, 2, "CALL pschgmtr"
betton selecais, 2, 2, "CALL pentrmis"
```

```
button
                    modeumstr, 4, 2, "CALL dummy" specuration 6, 2, "ADD_WINDOW normalism 13 3",
                                         "VPDATE DATCH BLEBORD",
                                         "CALL MEREANCE Screen Window Datum Sutton",
                                         "CALL stropy oldscreen Screen",
"CALL permise cid.arr"
      }
   WINDOW mtraumoom
      2000
                      78
      1120
                      2,0, 2,77
                      mtraem, 0, 2
mtrdesc, 1, 2
                   comi (
  WINDOW mtrans
     WOZ_MPE
      awa_columns
                      3,0, 3,77
                      mtraem, 0, 2
mtrdess, 1, 2
      deter
      deten
                      mismonel, 2, 2
             Declarations for DEFINE/MODIFY SCREEN (defnodetrscreen)
  DATOM prelomalt (
       num_rows 1
num_columns 10
       Variable prelowalt, 0,1
pickable MO
 DATOM ceriowalt (
       News and
       num_columns
                      29
                       curiowalt, 0, 16
       variable
       leader
                       "Low altitude:
       belpfile
                       help/curloult.hlp
 DATOM profistype (
       NUM_FORM
       awn_oolumns 13
       variable prefixtyp, 0,0
      pickeble
 DATOM curfixtype (
       FEET TOWS
       atta_columns 24
       variable
                    curfixtyp, 0,11
"Fix type: "
help/curfixtp.hlp
      leader
      helpfile
    Declarations for entering coordinates
DATM entlong {
    sum_rows 1
    awm_columns 25
    variable entlong, 0 ,11
    leader "Longitude:"
    helpfile help/markmap.hlp
DATOM entlet (
      num_rows 1
     num_columns 25
variable entlet, 0 ,11
leader "Letitude: "
helpfile help/markmap.hlp
- Declarations for show coordinates
```

```
DASTOM shwlat (
         Rum_rows 1
         variable shwimt, 0,0
pickable so
helpfile help/combmep.hip
   DATCH shwlong {
         aum columns
                        14
        variable shwlong, 0 ,0
piokable NO
helpfile help/combmap.hlp
   DAFOM profindist (
        RUM DOUGH 1
        variable prefindst, 0,4
pickable NO
  DATOM ourfindist (
       Num Lone
        awa columns
                      26
                       ourfindist, 0,16
        Verieble
                     "Fiz distance: "
        leeder
                       help/ourfiedi.hlp
        belpfile
  DATCH prefixed (
       BIR FOW
        num_columns
       variable prefixed, 0,4
       pickable
                      100
  DATOM carfixred (
       BWE_FOWS
       RWM Columns 26
       Variable .
                      ourfixed, 0,16 "Fix radial: "
       leader
       helpfile
                   help/corfiers.hip
 DATM prefixed (
aum_rows 1
aum_columns 10
variable prefixed, 0,4
pickable 20
 DATOM ourfleid (
      RUM_rows 1
      num_columns
                     curfixed, 0,16
      verieble
      leeder
                      "Fiz ID:
      bolpfile
                     help/curfixed.hlp
DATOM pressypt (
     aum_rows 1
aum_columns 10
variable premavpt, 0,4
pickeble 20
DATOM GERMANPE (
     BW POWE
     PAR COLUMN
                   24
     Variable
                    ournawpt, 0, 16
"Hav. Point: "
help/ournawpt.hlp
     leader
     belpfile
DATOM Bountres (
     RUE FOWS 1
                     soutre, 0,7
```

```
helpfile
                    help/stretrem.hlp
BUTTOW mestacopt (
     aum_rows 1
aum_columns 35
      textline nextsavptxt, 0, 2
helpfile help/xxtxxv.hlp
TEXTLEM retairestric ("Continue without selecting MER")
TEXTLEM retairestric ("Continue without greating mission")
TRITLINE retairestat2 ("Save mission in database")
TRITLINE retains ("Continue without selecting mission")
SUFFOW retmirent (
   ava_rows 1
   awa_columns 40
   textline retatreetxt, 0 ,2
belpfile belp/sessuatr.hlp
MUTTOW returnet1 (
   Sam Load
   ava_oolumns 40
   textline retairemixt1, 0 , 2 helpfile help/nomiss.hlp
MUTTOW retaitment2 (
   htm_rows 1
   Rum columns 30
   textline retutrent::2, 0 , 2 helpfile help/seveniss.hlp
NOTICE retnomis (
   NEEL LOAD 1
   aum_columns 40
textline retnomis, 0 , 2
helpfile help/noneumis.hlp
WINDOW definedatz
      Bum_ross
       ave_oolumes
                      78
       textline
                      ourpretzt,
                                      0, 16
      textline
                      ourpretzt3,
                                     1, 16
       datus
                      ournevpt,
                                      2, 1, "CALL VCAPITAL Sourmevpt",
                                               "HENVALA"
       detun
                                     2, 26
                     pressypt,
                                     3, 1, "CALL VCAPITAL Sourfield".
       detu
                      ourfixed.
                                                "MENVALS"
       deten
                     prefixid,
                                      3, 26
       detu
                      ourflared,
                                      4, 1
       detu
                      prefirred,
                                      4, 26
                      ourfirdist,
       detun
                                     5, 1
       detun
                     prefindist,
                                     5, 26
                                     2, 38, "CRLL lat2dec Sent",
      detun
                      shwlat,
                                     2, 64
      detun
                                     3, 38, "CALL lon2ded Sent",
                     estlong,
                                               "BEWVALE"
                                     3, 64
      deten
                     shwlong.
                      curfixtype,
                                     4, 38, "CALL VCAPITAL &curfixtyp",
                                               "MENVALS"
      detun
                     prefixtype,
                                     4, 64
      detu
                      curlowalt,
                                     7, 2, "CALL altidec &curlowalt",
                                               "MENUALS"
      detre
                     prelowalt,
                                     7, 26
                     ourhighalt,
                                     0, 2, "CRIL elt2des &gurhighelt",
                                               "MEWVALS"
      detun
                     prehighelt,
                                     8, 26
      detun
                     ourwidleft,
                                     9, 2
      detus
                     prewidleft,
                                     9, 27
      detum
                     curvidright, 10, 2
prewidright, 10, 27
```

```
ourartes,
                                              11, 2, "CALL VCAPITAL Sourertos",
                                                               "MENVALS"
                              presented, 11, 26
sentempt, 13, 2, "CALL astatrpt"
   NOTION cascalate (
         Of CARCALMENT (
REM_rows 1
REM_columns 35
textline cancelmentxt, 0, 2
helpfile help/canciment.hlp
  NOTICE SEVENTY (
        ava_rows 1
         textline seventrixt, 0, 2
helpfile help/seventr.hlp
  WINDOW mirection (
        num_rows 2
num_columns 78
line 0,0,0,77
button seventr, 1, 2, "CALL seventr"
button canonistr, 1,40, "CALL concentr"
  SCHOOL definodatracrees (
        title "DEFINE/MODIFY MER"
window stramoon, 2, 1
window definodatr, 6, 1
window straotion, 20, 1
border YES
                    Declarations for SELECT ANOTHER MER SCHOOL
  BUTTON shownerest: (
      numrous 1
numrolums 37
tertline showncustrum, 0,2
helpfile help/showntrs.hlp
 BUTTOM staematz (
        awa_rows 1
awa_columns 40
       textline stmentrtxt, 0 ,2
helpfile help/startxtr.hlp
 WINDOW memetram (
    aum_columns 40
datum neustram, 0, 2
WINDOW strecalmtr (
    Man_rows
     htm_rows 4
htm_columns 78
                        strountr, 1, 2,
                        "ADD_WINDOW novembros 6 3",
                        "UPDATE DATOM DOWNTERS",
    betton showners n2bv.arr*
betton shownerstr, 1, 41,

"CALL mebunch"
tartline recallstrtrt, 3, 2
WINDOW Gurdet (
aum_rows 10
aum_columns 32
     datum muldatoo, o, o
     datum muldat01, 1, 0
     detum muldet02, 2, 0
     datum muldat03, 3, 0
     detum muldet04, 4, 0
     detum muldet05, 5, 0 detum muldet06, 6, 0
```

```
datum muldato7, 7, 0
      datum muldatos, s, o
      detum muldatos, s, o
  WINDOW cardet1 {
                   10
      BUR FOWS
      num_columns 32
      detum muldet10, 0, 0
      datum muldatil, 1, 0
             muldet12, 2, 0
      detum muldet13, 3, 0
      detum muldet14, 4, 0
      datum muldat15, 5, 0
      datum muldet16, 6, 0
      detum suldet17, 7, 0
      datum muldatio, 0, 0
      datum muldat19, 9, 0
 WINDOW Guintibut (
      RUM COLUMNS 2
      button mulbut00, 0, 0, "CALL Milkoons deplault[0].arr"
     button mulbut01, 1, 0, "CALL MTRooms deplmult[1].arr"
button mulbut02, 2, 0, "CALL MTRooms deplmult[2].arr"
      betton mulbeto3, 3, 0, "CALL MITROORS deplement[3].arr"
     betton mulbet04, 4, 0, "CALL MTRooms deplimit[4].arr"
betton mulbet05, 5, 0, "CALL MTRooms deplimit[5].arr"
     bettoa mulbet06, 6, 0, "CALL MFROGAN deplault[6].arr"
bettoa mulbet07, 7, 0, "CALL MFROGAN deplault[7].arr"
bettoa mulbet08, 8, 0, "CALL MFROGAN deplault[8].arr"
     button mulbut09, 9, 0, "CALL MIROOMA deplacit[9].arr"
 WINDOW
           cumutrbet1 {
                 12
     BUEL FORM
      2001
     button mulbut10,
                            0, 0, "CALL MIROCON depleuit[10].arr"
                           1, 0, "CALL MTRooms deplault[11].arr"
2, 0, "CALL MTRooms deplault[12].arr"
     button mulbutil,
     button malbut12,
                            3, 0, "CALL MITROGAN deplault[13].arr"
4, 0, "CALL MITROGAN deplault[14].arr"
     buttom mulbut13.
     button mulbut14,
     button mulbut15,
                            5, 0, "CALL MIROCON deplouit[15].arr"
     button mulbet16,
                            6, 0, "CALL MIROCOR deplault[16] arr"
     button mulbut17,
                           7, 0, "CALL MTRooms deplanit[17].arr"
                            8, 0, "CALL MIROONA deplacit[18] arr"
     button mulbut10,
                            9, 0, "CALL MIROCAR deplement [19] arr
     buttom mulbut19,
WINDOW retmirent (
     THE TOWN
     awn_columns 45
                retairent, 0, 1, "CALL melisto",
                                      "CALL postrest
SCREEN obgommetreoreen (
      title "SELECT ANOTHER MER"
      Window mtrasmoom, 2, 1
      window strecalmtr, 5, 1
      window curntribut, 9, 3
window curntribut1, 9,42
      window curdet, 9, 5
      Window ourdet1, 9,44
      window retmirent, 20, 2
      horder TES
                          Declarations for new mtr name
DATOM detepubl (
     num_rows 1
      verieble
                     datepubl, 0, 22
     leader
                     "Date of publication: "
     helpfile help/dstepubl.hlp
MOTTOW definest: {
```

```
num_rows 1
num_columns 40
       textline definementate, 0 ,2
helpfile help/getmap.hlp
 WEMDOW defaunt: {
        awa_columns
                         77
         dett
                          origatr,
                                        0, 1
                         schedule, 2, 1
         detun
                         srods, 4, 1
datepubl, 6, 1
definemtr, 8, 1,
                                        4, 1
         detun
         deten
         button
                                                        "CALL estatspt"
  SCREW mirdefinescreen (
       title "META DEFINITION"
                   mtraemoon, 3,1
defauntr, 7,2
       window
       wladow
                   defauntr,
       window
                    mtractica,
                                        20,1
       border
                    THE
                     Declarations for MISSION RECUIREMENTS WINDOW
 VARIABLE ec_is_form (
type DETECTION
       type Diraces.
       lowlimit
                       1
       eplimit
                     16
2
       default
 VARIABLE perstant (
    foundia "VARCHAR pr_per_u.arr"
    type STRING
     type
format
                  4-6a
 DATOM perstuat (
    ava_columns 10
                    pwrstust, 0, 2
     verieble
    leader
    trailer
    pickable
VARIABLE mistype {
type string
formet $-is
DATOM mistypel (
     AUD FORM
      aum_columns 18
      variable mistype, 0, 14
leader "Mission type:"
helpfile help/mistype.hlp
DATOM mistype (
     num_rows 1
num_columns 16
variable mistype, 0, 14
lander "Mission type:"
pickable 20
VARIABLE presitiev
foundin "
                "ALTSTEC prelowalt.spec"
STRING
196
      type
      formet
DATOM presition {
    num_rows 1
num_columns 10
variable presitiev, 0, 0
```

```
pickable
  VARIABLE curaltiev (
foundin "ALTEFEC curlowalt.spec"
      type
format
                     STRING
                     490
  DATOM oursitler (
       BAR LOAD
      aum rootume 21
variable caraltley, 0, 11
'ander "Alt:
       helpfile
                    help/altmtr.hlp
  ANTINETS selbes bas (
       type
format
                      DOUBLE
                      412.312
       lowlimit
                     0.0
4000.0
      uplimit
default
                     100.0
      M preparation 1
SME rows 1
Aum_columns 12
variable ac pre pwr, 0, 0
 DATCH prepurset (
 DOUBLE
                     $10.31£
                    0.0
      lowlimit
      wlimit
                     4000.0
      default
                     100.0
   }
DATOM curporest (
      hum_rows 1
hum_columns 21
      variable as our pwr, 0, 11
leader "Power:
      leeder
      helpfile
                     help/pwrspec.hlp
 DATCH prespeed (
      num_colum
     variable ad pre spd, 0, 0
trailer "Ers"
pickable 80
VARIABLE ad_our_spd
   type
   formet
lowlimit
                  434
                  100
   uplimit
                  600
   default
                  450
DATUM ourspeed (
     aum columns 21
variable ad columns
                   ad_our_spd, 0, 11
"Speed: "
"Ers"
     trailer
     helpfile
                    help/strspeed.hlp
DATOM presevpt1 (
    num rows 1
     veriable
```

pressypt, 0,0

```
piskable 20
 DATOM GERBAUPt1 (
      ava columns 21
       variable curnavpt, 0, 11
leadar "Envpoint:"
helpfile help/navame.hlp
       leader
      helpfile
 TEXTLES canciparatet ("Cancal data entry for this mission")
 MOTTON cencipara (
      sum_rous 1
sum_cous 35
textline onsciperatit, 0,
helpfile help/ossdistr.hlp
                       ommolperatut, 0, 2
 THITLINE saveparetzt ("Save this mission's data")
 NOTICE SEVERALE (
      RWE FOW
      aum_columns 30
      tartline saveparatut, 0, 2
helpfile help/saveniss.hlp
                      saveparatzt, 0, 2
TEXTLINE operdstatzt ("Ready to enter operational data")
NUTTON operdata (
    num rows 1
      textline operdstatxt, 0, helpfile help/estrope.hlp
                       operdstatzt, 0, 2
TEXTREOCK str (
    aum_rows 10
aum_columns 36
     filenene
                 tatblk/mtr.tat
TEXTLEM marlabeltat ("MAV FIX
                                                FIX TIPE
                                                                WIDTE" )
TEXTLINE strict ("REFERENCE INFORMATION FOR DATA SHIRY")
TEXTLINE ourpretati ("COMMENT PREVIOUS")
TEXTLINE curpretat2 ("----")
WINDOW operatry (
   aum rows 10
aum columns 78
button columns
  EGM_ LOAD
  button operandy, 2, 19, "CALL penddope 1"
button operando, 7, 19, "CALL penddope 12"
WINDOW fitpers (
      aum rows 17
aum columns 70
      detun
                     mistype,
                                    0, 2
      deten
                     ECRESO,
                                    1, 2
      textline
                     mtrtmt,
                                    0,40
      textline
                     mtrlabeltzt, 2,38
                     cuspretxt1, 3,14
cuspretxt2, 4,14
cuspretxt2, 5,1, "CALL VCAFIZAL Scusparpt",
      textline
      textline
      detun
      detun
                     pressypt1,
                                    5.25
      detun
                     curspeed,
                                    7, 1
7,25
      detw
                     prespeed,
      detun
                      curpwreet,
      detus
                     preparent,
                                    9,22
                     pwrstunt, 10, 15
curaltlev, 12, 2, "CALL alt2dec Sourlowalt",
      detu
      detus
                                            "MENVALE"
      detus
                                  12,23
     boz
line
                     1, 37, 14, 77
                     3, 38, 3, 76
15, 0, 15, 77
```

```
narinavpt, 14, 2, "CALL arinispt"
operdeta, 16, 2, "MAW_SCREEN opentry"
cancipara, 16,40, "CALL cancais"
        betton
        betton
        textblock
 SCREEN strilt (
title "FLIGHT PARAMETER BETTY"
      title
Window
                   Miramoon,
                                    2,1
       window
                   fltpere,
                                    B, 1
 SCREEN opentry (
title "FLIGHT OPERATION DATA ENTRY FOR MER-
window stramson, 3,1
                   miramoon, 3,1
       window
                   opentry,
                                    6,1
                 Declarations for seasonal MFR and REQUIREMENTS
 DATTM decmite {
      num_rows 1
       num_columns 6
      variable decmits, 0, 0 helpfile help/methnite.hlp
   }
DANTOM · decday {
     Bum rows 1
      ave columns
      variable decday, 0, 5
leader "DEC:"
     helpfile
                     help/mathday.hlp
DATOM movaite (
   RMM_rows 1
     DATUM novday (
aum_rows 1
aum_columns
      wariable movday, .,
landar "MOV:"
-- refile help/mathday.hlp
                    10
DATOM octaite {
     aum_rows 1
      aum columns 6
      variable outnite, 0, 0 helpfile help/muthaite.hlp
     belpfile
DATOM octday (
     ave columns 10
      variable octday, 0, 5
leader "OCT:"
     leader
                 help/mathday.hlp
     belpfile
DATOM sepaite {
    zum_rows 1
     num_columns (
     variable sepuite, 0, 0 helpfile help/mathnite.hlp
DATOM sepday {
    sum_rows 1
     aum_columns 10
     variable
                   sepday, 0, 5
     leeder
     helpfile
                    help/mathday.hlp
```

```
DATOM auguite (
       BEE POWS 1
       aum_columns 6
       veriable
                     amgmite, 0, 0
       helpfile help/mathaite.hlp
  DATOM sugday (
       am_column
       verieble
                    augday, 0, 5
       leeder
       helpfile
                    help/mnthday.hlp
   }
  DATOM julnite (
num_rows 1
num_columns
                    •
       veriable
                    julmite, 0, 0
      belpfile
                    help/mathaite.hlp
   }
 10
      variable julday, 0, 8
leader "JUL:"
      helpfile
                 help/mathday.hlp
 DATOM jumnite (
      num_rows 1
      variable
      num_columns (
                   junnite, 0, 0
help/muthaite.hlp
     helpfile
 DATOM junday (
      ava_columns 10
      veriable
                   junday, 0, 5
      leader
                 help/mathday.hlp
      belpfile
 DATOM maynite (
    aum columns 6
     variable maynite, 0, 0
helpfile help/mnthmite.hlp
 DATOM mayday (
     atm_rows 1
     aum_columns 10
     variable mayday, 0, 5
leader "MAX:"
     belpfile
                  help/mathday.hlp
DATUM apraite {
     Man_columns (
                   apraite, 0, 0
     verieble
              apraite, u, w
help/mathaite.hlp
     helpfile
DATOM aprday (
aum_rows 1
aum_columns 10
     variable aprday, 0, 5
leader "AFR:"
helpfile help/mnthday.hlp
     leader
    helpfile
  }
DATOM marmite {
    RUM_rows 1
    Atm_columns (
    variable
                  marmite, 0, 0
    helpfile
                  help/mathaite.hlp
DATOM marday {
```

```
aum columns
                   marday, 0, 5
     variable
     leader
     helpfile
                   help/methday.hlp
DATCH febaite (
    RWA COLUMN
     variable
                   febuite, 0, 0
     belpfile
                   help/mathaite.hlp
DATOM febday (
     aum_rous 1
     Sam Coolamas
                   10
     veriable
                   febday, 0, 5
     leader
    bolpfile
                   help/mathday.hlp
DATUM jennite (
zwm_rows 1
     aum columns
variable
                   jeamite, 0, 0
                   help/methnite.hlp
     halpfila
DATOM jendey {
     SAME COLUMNS
                   janday, 0, 5
     veriable
     leader
                help/mathday.hlp
     helpfile
WINDOW mosth (
     aus rows 14
    hum columns 77
tarkline instrict, 1, 1
tarkline dayaite, 3, 0
datum janday, 4, 1
     detum jennite,
                         4, 12
     datum febday,
                         5, 1
     detum febmite,
                        5, 12
     datum marday,
                         6, 1
     datum marmite,
                         6,12
     datum aprday,
                         4.20
     detum apraite,
                         4,31
     detum mayday,
                         5,20
     dete
            mayaite,
                         5,31
     datum junday,
                         6,20
     datum jummite,
                         6.31
     detun
            julday,
                         4,39
     detum julmite,
                         4,50
     detun
            augday,
                        5,39
     detum augmite,
                        5.50
     detum sepday,
detum sepaite,
                        6,39
                        6,50
     datum octday,
                        4,58
     deton
            octaite,
     datum movday,
                        5,50
     detum acvaite,
                        5, 69
     datum deadey,
                        6,58
     detum decaste,
                        6,69
     button savepara, 13, 2, "CALL savemis"
     button cancipara, 13,40, "CRIL cancais"
    Declarations for new SELECT AIRCRAFT AND MISSION FOR MER
TEXTLES shemoremistri ("Show more mission names (if any)")
BUTTON shumoremis (
    NO. LONG
                  1
30
    num_columns
     textline
                    shumoremistrt. 0. 2
    helpfile
                    help/sharmiss.hlp
```

```
TEXTLIRE recalmis ("Recall one of the following missions")
  WINDOW Downlaws (
     RIES_FOWS
     avan_columns 40
                   mismana, 0, 2
  WINDOW CRIMICS {
      PART TORR
      num_columns 78
                   معد,
                           1, 2, "ADD_WINDOW normicsm 7 3",
                                 "UPDATE DATCH SIGNAS",
                                 "CALL penumies old.arr"
     button shunoremis, 1, 40, "CALL sebunch"
     textline recelmis, 3, 2
      datum muldatoo,
                           4. 4
      detum muldat01,
                           5, 4
      detum muldet02,
                           6, 4
      detum muldetos,
                          7, 4
      detum muldet04,
     datum maldetos.
                             4
      datum maldatos,
                          10, 4
     datum muldet07,
                          11, 4
     datum amildatos,
                          12, 4
     detum muldetos.
                          13, 4
     datum muldet10,
                          4.42
     detum muldetil,
                          5.42
     detum muldet12,
                          6.42
     detum muldet13,
                          7,42
     detum muldet14,
                          0,42
     detum muldet15,
                          9, 42
     datum muldet16,
                          10,42
     datum muldati7,
                          11.42
     datum muldet10,
                          12.42
     detum muldet19,
                          13,42
     button mulbut00,
                          4, 2, "CALL MISCORE deplet[0].arr"
     button mulbut01,
                          5, 2, "CALL MISCORN deplault[1].arr"
     button mulbut02,
                          6, 2, "CALL MISCORN deplault[2].arr"
7, 2, "CALL MISCORN deplault[3].arr"
     button mulbut03,
     button
            mulbut04,
                          8, 2, "CALL MISconn deplault[4].arr"
                          9, 2, "GALL MISCORN deplanit[5].arr"
     button
             malbut05,
     buttom mulbut06,
                          10,2, "CALL MISCORE depleuit[6].arr"
            malbat07,
     button
                        11, 2, "CALL MISCORE depleuit[7].arr"
     button mulbut08,
                        12, 2, "CALL MIScoun deplanat[8].err"
     button malbutos,
                        13, 2, "CALL MISCORN deplement[9].arr"
     button mulbet10,
                          4,40, "CALL MISCORN deplault[10].arr"
     button mulbutil,
                          5,40, "CALL MISCORN depleuit[11].arr"
                          6,40, "CALL Misconn deplanit[12] arr"
     button mulbut12,
     button mulbut13.
                         7,40, "CALL MISCORN deplement[13] .azz"
     button mulbut14,
                         8,40, "GALL MISCORN deplault[14].arr"
     button mulbet15,
                         9,40, "CALL MISCORE deplacit[15].arr"
     button mulbut16,
                         10,40, "CALL MIScoon deplault[16] arr"
                        11,40, "CALL MISconn deplanit[17].arr"
    button mulbut17,
    button mulbut18,
                        12,40, "CALL MISCOON deplanit[18] .arr"
    button malbut19,
                       13,40, "CALL MISCORN deplanit[19] arr"
15, 2, "CALL selisto",
    button retnomis,
                               "CALL MEN SCREEN oldscreen"
TEXTLINE selmisstat ("Select mission")
BUTTON selmies (
     aum columns
                    38
     textiine
                    selmisstat, 0, 2
     helpfile help/selmiss.hlp
THEFAUR fitperinftst ("Specify flight permeter information")
NOTION fitperiaf (
     PAR LOAD
                 -
37
     AUM Golyman
     textline
                   fltperinftxt, 0, 2
     helpfile
                  help/fitperem.hip
WINDOW selection {
                       15
    PAR LOAR
    ava columns
                       77
     datum accemel,
                      1,1, "CALL vfysomer tid.arr"
```

```
button selmins, 3,2, "CALL peolymis"
button filtparinf, 5,2, "CALL pentrfilt"
      button retairment1, 13,2, "CALL canonis"
SCHEEN spectrais (
title "SELECT AIRCRAFT AND MISSION FOR MER"
                  stramocmi, 3,1
selecties, 7,1
      wladow
      window
      border
                   TES
SCREEN objects (
*SELECT AIRCRAFT AND MISSION FOR MER-
      window
                 mtrassconi, 2,1
      window cumiss, 6,1
      border
                   TES
       Declarations for Day-Wight window
DATOM alght (
      ME FOWS
      awn_columns 16
      variable might, 0, 8
leader "Wight:"
      belpfile
                   help/miteops.hlp
DATOM day (
      aum columns 16
      variable day, 0, 6
leader "Day:"
      leeder
      belpfile
                   palp/dayops.hlp
WINDOW dayaite {
    aum_rows 12
aum_columns 77
      datum acmame, 1, 1
textline instrictut, 3, 1
      datum day, 5, 2
datum night, 5,15
button savapara, 10, 2, "CALL expmope",
"CALL savanis"
                                       "CALL savenis"
      button candipara, 10,40, "CALL canonis"
border YES

    Declarations for mission specification window

DATUM sortie (
     Bun_rows
     num_rows .

num_columns 37

variable sc_im_form, 0, 35

leader "Number of aircraft in formation: "
helpfile help/numform.hlp
VARIABLE misdesc (
   foundin "VARCHAR misdasc.arr" -this description is for mission
   type STRING
format 4-60s
DATOM misdesci (
   ava_rows 1
ava_columns 76
   variable misdesc, 0, 7
leader "Descr:"
pickable MO
DATOM misdess (
   REM FOWS
   num_rows 1
num_columns 76
variable misdesc, 0, 14
```

```
"Description:"
       helpfile
                    help/misdess.hlp
   WINDOW strassocal (
      BUR TOWN
                    . 78
       aum_columns
      معدد
                       2,0, 2,77
                     misnamel, 0, 2
misdesal, 1, 2
       detus
      )
  WINDOW misspec (
                      16
77
      REEF LORS
      aum columns
detum misdesc, 1,1
detum mistype1, 3,2
evetie, 5,2
      aum columns
      datum sortia, 5,2
button retstrenti, 14,2, "CALL NEW SCHEEF oldscreen"
button retstrent2, 14,47, "CALL entermis"
    }
  SCHEEN missped (
title "MISSION SPECIFICATION FOR AN MER"
        window
                    stramoom3, 3,1
                  misspec, 6,1
        Window
        border
  -01-14-88 st will try to fix this file so it wil be only for assessments -01-14-88 st changes will be commented with date
  -01-16-88 st change title on first screen, change lat. long. format lon
  - map control
-01-17-88 st con't to fix sdf file commented with 1-17-88
  - DECLARATIONS FOR SCREEN MEADER
  VARIABLE assessme (
foundin "ASANTEADR ASSESSMENT.name"
     type STRING
format 4-30s
 DATOM assessme (
      num_rows 1
num_columns 50
      variable assessment, 0 , 18
leader "Assessment name: "
pickable NO
 DATOM essensed {
     ava_rows 1
ava_columns 75
      variable assemme, 0 , 30 leader "Home of current assessment: "
     leeder "Ne
picksbie NO
 VARIABLE comment (
     foundin "ASAMEZADR ASSESSMENT desc"
     type STRING
format 4-66s
DASTM comment (
    num_rows 1
num_columns 76
     variable comment, 0 , 10 leader "Comment: " pickable 20
VARIABLE Gurmapham (
     formet 4-30s
DATOM Gurmapman (
```

```
aum_columns 77
Variable gurnspuss, 0 , 19
leeder "Current map asse: "
      pickable
  WINDOW lin (
     aum ross 1
aum columns 78
lime 0, 0, 0,77
 WINDOW assumed (
      Par Load
      num_columns 78
                assemmel, 0,1
comment, 1, 2
2, 0, 2,77
      detus
 WINDOW assumedia (
      ama_rows 4
      awn_column
                     essemmel, 0, 0
comment, 1, 1
      deten
                     ournepass, 2, 1
                    3, 0, 3,77
    Declarations for buttons used in majoraction footer
 TEXTLINE probetatit ( "REVIEW CURRENT ASSESSMENT STATUS" )
 MUTTOW probetet (
      RUR FOWS
      atm_columns 35
     taxtline probstatxt, 0, 2
helpfile help/probstat.hlp
 TEXTLINE probetatit2 ( "REVIEW ASSESSMENT STATUS" )
 NOTION probetat2 (
NOTION 1
NOTION 26
     tastline probstatut2, 0, 2
helpfile help/revstat.hlp
 TEXTLEM Probded ("ADD INFORMATION TO CONSERT ASSESSMENT" )
 BOTTOM probdef (
     awa_rows 1
     num_columns 39
     textline probdef, 0, 2
helpfile help/probdef.hlp
TEXTLINE probdef1 ("ADD TO ASSESSMENT DEFINITION" )
NOTION probdef1 {
    BWE_FOWS
     awa_rows 1
zwa_columns 31
    textline probdef1, 0, 2
helpfile help/probdef.hlp
THISTLINE enclysis ( "AMALYEE DATA" )
BOTTOM analysis (
    sum_columns 15
    textline enalysis, 0, 2
helpfile help/dstenel.hlp
TEXTLINE report on { "MAKE A REPORT" }
BUTTOM reportgen {
    NUM_ NOWS
```

```
testline
                         reportgen, 0, 2
        helpfile
                         help/reportgen.hlp
  TEXTLE WORKLISTER ( "VIEW CERCELIST FOR CURRENT ASSESSMENT" )
  BUTTON weakhlist (
       PAR LOAD
       awa columns 42
       textline vechklistst, 0, 2
helpfile help/viewlist.hlp
            Declarations for majoraction footers
  WINDOW majorastics (
                                          -header for problem status sure
       REEL FORM
       ave columns
                       78
       محدد
                          0, 0, 0,77
       title
                         "Alternative actions you can now take:"
                        problef, 2, 2, "MEW_SCREEN problefsureen"
analysis, 2,56, "MEW_SCREEN analysisureen"
weaklist, 3, 2, "MEW_SCREEN viewchecklistsure
reportges, 3,56, "CALL dummy2"
       button
       button
       button
       button
   3
  WINDOW majorastical {
                                       -header for problem definition screen
       BWE TOWS 4
       awa_columns 78
       1120
                    0, 0, 0,77
       title
                     "Alternative actions you can now take:"
      button analysis, 2, 2, "MEW SCREEN analysiscreen"
button probutat2, 2,48, "CALL perrobut"
button venhilist, 3, 2, "MEW SCREEN viewchecklistscreen"
button reportgen, 3,48, "CALL dumy2"
 WINDOW majoraution! {
                                       -header for data analysis screen
      MAN TOWN
       atta_columns 78
       معند
                     0, 0, 0,77
                     "Alternative actions you can now take:"
probdef1, 2, 2, "MEW SCHEEN probdefsurees."
      title
       button
                    reportgen, 2,47, "CALL dummy2"
venklist, 3,2, "HEW SCREEN viewchecklistscreen"
probstst2, 3,47, "CALL psprobst"
      betton
      button
      button
  }
 WINDOW majoraction3 { "header for report generation surses
      NW TOWS
      ava_columns 78
      1120
                      0, 0, 0,77
                      "Alternative actions you can now take:"
      t1tle
                     venthist, 2, 2, "MEN SCREEN viewchecklistscreen"
probstat2, 2, 47, "CALL peprobst"
probdef1, 3, 2, "MEN SCREEN probdefscreen"
analysis, 3, 47, "MEN SCREEN analysiscreen"
      button
     button
      button
 }
WZEDOW majorastica4 {
                                     -header for view checklist screen
     NO POWE
      awa_columns 78
     1486
                     0, 0, 0,77
     +4+1-
                     "Alternative actions you can now take:"
                     probetti, 2, "CALL peprobet"
reportgen, 2,47, "CALL dummy?"
probdefi, 3, 2, "HEW_SCRIEN probdefscreen"
analysis, 3,47, "HEW_SCRIEN analysiscreen"
     button
     button
     button
TEXTLINE loadglobal ("LOAD LOCAL DATA FOR GENERAL ACCESS")
BUTTOW loadglobel {
   PAGE TOME
   num columns 39
   textline
                 loedglobel, 0, 2
help/loedglob.hlp
   helpfile
```

The second second second

\_....

num columns 16

```
WINDOW majorastica5 (
                                       -footer for detabase housekeeping squeen
         BEEL TOWN 4
         aum_columns 78
                 0, 0, 0,77
         Line
        title "Alternative actions you can now take:"
betton loadglobal, 2, 2, "CALL demay"
button assessment, 3, 2, "CALL peprobet"
   SCHOOL MITTERY (
       title "MER DATA BETRY"
       window mtramnoom, 3,1
window mtratestry, 7,1
window majorantion3, 18,1
border YES
                      Declarations for problem status screen
   VARIABLE startdate (
       foundin "VARCEAR startdate.arr"
      type symmet
format 4-21s
  DATOM startdate (
       htm_rows 1
       htm_columns 40
      variable startdate, 0 ,15 leader "Date started: " pickable 20
  VARIABLE lastdate (
      foundin "VARCHAR lastdate.arr"
type STAIRN
format 4-21s
 AUM_rows 1
AUM_columns 55
variable lastdate, 0 , 28
lasdar "Date of last modification: "
pickable mo
}
  DATOM lastdate (
 VARIABLE plantlast (
     foundin "VARCHAR plannist.arr"
type STRING
format 4-20s
 DATOM plearlest {
variable plantlast, 0 ,19 leader "Last modified by: "pinkable No
     ham rows
WIRDOW probetst (
     Dan Lone
     RVE COLUMNS 78
     detus
                 startdata, 0, 2
lastdata, 2, 2
planriast, 4, 2
     deten
TEXTLINE chquestrt ("Nork on a different assessment")
MOTTON chars (
    AUM rows 1
```

-change to different assessment

WINDOW abgrasess (

```
num_columns 78
                      chgass, 0, 2, "CALL pschgass"
              Declarations for ENVIRONMENTAL ASSESSMENT DEFINITION SCHOOL
   TEXTLINE modeostat ("Nork with MDA information (number or type of aircraft, missions, etc.)"}
  BUTTON modmon (
       ava_columns 76
       textline modmostxt, 0 ,2
helpfile help/moswork.hlp
  TEXTLEM modetrixt ("Nork with METR information (number or type of aircraft, missions, etc.)")
  NOTTON modestr (
       RUM FOWS 1
       REAL COLUMNS 76
      tertline modetrtzt, 0 ,2
helpfile help/strwork.hlp
  TEXTUINE modesptxt ("Nork with map information (designate land uses, update maps)")
  NUTTON modmap (
       mum_oolumns 76
      textline modmeptxt, 0 ,2
helpfile help/mepwork.hlp
  TENTLINE selectri ("Actions you can now take to add information to this assessment:")
  WINDOW selection (
      BUR FOW
      aum_columns 77
      textline selectat, 0, 1
                   modmap, 2, 1, "CALL dummy2"
modmir, 4, 1, "CALL pentrent"
modmon, 6, 1, "CALL dummy"
      button
      button
     button
 SCREEN probdefsdreen {
    title "ENVIRONMENTAL ASSESSMENT DEFINITION"
    window assumation, 7, 2
    window majoractioni, 18, 1
    border YES
          Declaration for entering coordinates window
 TEXTLINE estupplf
   {"Enter upper-left corner coordinates of area of current interest"}
TEXTLINE entlowet
   {"Enter lower right corner coordinates of area of current interest"}
DATOM shwlat2 (
      BUR FOWS 1
      aum columns
                      25
     variable shwist, 0 ,11
leader "Letitude: "
helpfile help/combmsp.hlp
DATOM shwlong2 (
     num rows 1
                     25
      variable shwlong, 0 ,12
leader "Longitude: "
      leader
     helpfile help/combmap.hlp
WINDOW entocor {
```

```
num_cons
                     10
                   78
                    estupplf, 2, 2
                    entlat, 3, 1, "CALL lat2ded Sent", "HEWVALS"
entlong, 3, 38, "CALL lon2ded Sent", "HEWVALS"
                                         "CALL lat2des Sent", "MENVALS"
      deten
      testiine
                    estlowst, 5, 2
                    shwlat2, 6, 1, "CALL lat2des &show", "MENVALS" shwlong2, 6, 38, "CALL lon2des &show", "MENVALS"
      dete
      deten
            Declarations for DATA ANALYSIS SCREEN
 TEXTLINE geodetinqtxt ("Make geodets inquiries on map surees")
 BUTTOM geodeting (
     BUR FOWS
      aum_columns 76
     textline geodatinqtxt, 0 ,2
helpfile help/geoinq.hlp
 TEXTLINE compacisaftat ("Compare noise effects") -1-17-00
 SUTTON companied (
                                               -1-17-88
     RUM_FOWS 1
                                               -1-17-00
     aum_columns 76
                                               -1-17-88
     textline compacisafixt, 0 ,2 helpfile help/compacis.hlp
                                               -1-17-00
                                                    -2-06-88
                                    -1-17-88
 TENTRIME calculatestat ("Calculate noise effects in specified area")
 MUTTOW calcactsef (
     RWM_FOWS 1
     BWB columns 76
     textline calcaciseftst, 0,2
helpfile help/effcalc.hlp
TEXTLES calcuoisemptri ("Calculate noise emposure in specified area")
 MUTTON calcactearp (
    NUM YOUR 1
     awn_columns 76
    taxtime calcacisexptxt, 0 ,2
helpfile help/exposic.hlp
TELETAIRE caloquiochtzt ("Calculate quicklook (point) exposure estimate")
BUTTOW calcoklook (
    MOT_ENVE
    num columns 50
    textline calcoptlocktxt, 0 ,2
helpfile help/qcklock.hlp
TEXTLINE estocortat ("Enter coordinates from keyboard")
SUFFOR extenor (
    AUR POWS 1
    aum_columns 40
    textline entocortxt, 0 ,2
helpfile help/kbdstry.hlp
TEXTLINE usemaptat ("Use map screen")
    amm columns 30 textline
                                  mesmaptat, 0 ,2
   helpfile help/usemap.hlp
TEXTLINE selectaratut ("Actions you can now take to analyze environmental assessment data:")
TEXTLINE defgeoerestri ("Specify a geographic area of interest:")
WIMPOW detamention {
    NUM_rows 11
```

```
ave columns
                      78
       textline
                      selectemetat, 0, 2
                                      2, 2, "CALL dumy"
      bettoe
                      omlogklook,
      button
                      celcacteerp,
                                      3, 2, "CALL dumy"
      betton
                      celcocisef,
                                       4, 2, "CALL dumy"
      button
                      compacisef,
                                       5, 2, "CALL dum
                      quodating,
      button
                                       6, 2, "CALL de
      testline
                      defgeommentst, 8, 2
                                      9, 2, "CALL dummy2"
10,2, "ADD_NIMBOW enteror 5 1"
      button
                      wenep,
                      estecor,
 SCHEME analysiscreen (
title "DATA MEMITES"
window assumncom, 2,1
window detanaction, 6,1
window najoractica2, 10, 1
border TES
 TEXTLINE showmoretet ("Show more assessments (if any)")
 BUZZON showners (
      aum_rows 1
aum_columns 37
      taxtime showeretxt, 0 ,2
helpfile help/getassunt.hlp
 TRITLINE recalls stat ("Recall one of the following assessments:")
 TEXTLINE stanwartat ("Start a new assessment")
 BUITON STROMAS (
      REE FORE
                     20
      aw_column
      textline
                       stmewasstat, 0 ,2
      halpfile
                      help/newssmat.hlp
VARIABLE newspam {
    foundin "VARCHAR m2bv.arr"
      type
formet
                  STRING
                  4-30e
DATCH newscam (
      PART LOAM
                      1
      Num_columns
                      40
      variable
                      novement, 0,7
      leeder
      helpfile
                      help/stacess.hlp
WINDOW NAMES OF (
     TAN LONG
                      1
      BUR COLUMN
                      40
                      newscam, 0,1
WINDOW stmewass {
     NAME TO ARR
     ava_columns
                      78
                      STROWNS, 0, 2, "ADD WINDOW ROWNSHAM, 6 3",
"UPDATE DATOM ROWNSHAM",
     betton
                                "CALL penwasen a2bv.arr"
     button
                      showners, 0, 41,
                                             "CALL ubunch"
WINDOW recalass {
    BWE FOW
     awa_oolusse
                     78
     testline
                      recallassint, 0, 2
         Geresbut (
      PART LOAM
       ava_columns
       button mulbut00, 0, 0,
                                       "CALL ASAMoonn deplement[0].arr"
       button mulbut01, 1, 0,
button mulbut02, 2, 0,
button mulbut03, 3, 0,
                                      "CALL ASAMoon deplault[1].arr"
                                       "CALL ASAMoonn deplanalt [2] .arr"
                                       "CALL ASAMOORE depletit[3].err"
```

```
button
                  mulbut04, 4, 0,
                                        "CALL ASASoom deplesalt [4] .arr"
         bettoe
                  malbut05, 5, 0,
                                        "CALL ASANGORA deplanit[5].arr"
         bettoe
                  malbut06,
                               6, 0,
                                        "CALL ASAMoonn deplement[6].arz"
                  mulbut07, 7, 0,
mulbut08, 8, 0,
mulbut09, 9, 0,
         button
                                         "CALL ASAMocen deplesalt[7].err"
         bettoe
                                         "CALL ASAMoona deplanit[8].arr"
                                         "CALL ASSESSED depimult[9].arr"
         button
 WINDOW cursebut1 {
         num_rows 10
         200 2000
         button melbut10, 0, 0,
                                       "CALL ASAMocea depleuit[10].err"
         button mulbut11, 1, 0,
                                        "CALL ASAMoons depleuit [11] .arr"
         button mulbut12, 2, 0,
                                       "CALL ASANGORE deplement [12] .arr"
         button mulbut13, 3, 0,
                                        "CALL ASAMoona deplault[13].arr"
         button mulbut14, 4, 0,
button mulbut15, 5, 0,
                                        "CALL ASAMooan deplault [14] .arr"
                                        "CALL ASAMoona deplement [15] .arr"
         button mulbut16, 6, 0,
                                        "CALL ASAMoona deplanit[16].arr"
         button mulbut17, 7, 0,
                                        "CALL ASSMOORN deplault[17].arr"
        button mulbut18, 8, 0,
button mulbut19, 9, 0,
                                        "CALL ASAMoonn deplault[18] .arr"
                                        "CALL ASAMooan depleratt[19].arr"
 TEXTLEE retnouss ("Continue without selecting assessment")
 BUTTON retessent (
    NAME LOAD
     awa_oolwans 40
    textline retnoss, 0 ,2
helpfile help/mossess.hlp
 WINDOW | retwoess {
    PART LOAM
                1
    num columns 76
                  retassent, 0,1, "CALL mista", "CALL peprobet"
 SCHEME chacures creen (
      title
                  "SELECT ANOTHER ASSESSMENT"
      Window
                  assamoom, 2,1
      window
                  strowess,
receless,
                               6.1
      window
                               8,1
      window
                  oursesbut, 9,3
      window
                  ouresebet1, 9,42
      Window
                  curdet,
                             9,5
      window
                  ourdet1, 9,44
retwoes, 20,2
      window
                  TES
   Declarations for NEW ASSESSMENT DEFINITION
 VARIABLE entdeedl (
      foundia "VARCHAR60 entdesc[0].azz"
     type STRING
format 4-6
                4-60e
DATCM emtdesdi (
     aum_rows 1
aum_columns 75
     variable entdess1, 0,0
helpfile help/entdess.hlp
VARIABLE entdess: {
     foundin "VARCHAR60 entdesc[1].arr"
                 FIRE
     type STRING
format 4-60s
DATOM estdesc2 (
     num rows 1
num columns 75
     variable entdesc2, 0 ,0
helpfile help/entdesc.hlp
VARIABLE entdess3 (
```

```
foundin
                    "VARCEAR60 estdess[2].arr"
                       ##R.186
         type
format
                     $-60s
  DATOM estdessi (
        num_rows 1
num_columns 75
        variable entdesc3, 0 ,0
helpfile help/entdesc.hlp
  VARIABLE entdesd4 {
   foundia "VARCEAR60 entdesd[3].arr"
   type STRING
       type STRING
format 4-60s
  DATOM entdess4 (
        aum_rows 1
aum_columns 75
        variable entdess4, 0 ,0
helpfile help/entdess.hlp
  TEXTLES entdescrit ("Please enter a brief description for this assessment")
  DASTIM DAMESTALL (
        aum_rows 1
aum_columns 77
   variable newspam, 0 ,25
leader "Rame of new assessment definition:"
pickable BO
  WINDOW mewasdess (
       AME rows 9
       detum newssess, 1,3
textline entdescript, 3,4
        detim
                    entdesci, 4,4
entdesci, 5,4
entdesci, 5,4
entdesci, 7,4
YES
        detu
        detun
        detus
        detun
       border
             Declarations for database howsekeeping screen
 TEXTLINE updateinfotzt ("Update information")
 NOTICE updateinfo (
num_rows 1
num_columns 50
       textine updateinfotxt, 0 ,2
helpfile help/updatein.hlp
TELTIME asstabletric ("Frint list of all columns in an assessment's tables")
 NOTION asstable2 (
      ava_rows 1
ava_columns 55
      tartline asstabletzt2, 0 ,2
helpfile help/asstable.hlp
TRITLINE asstabletzt1 ("Frint list of an assessment's tables")
MUTTOW asstable1 (
     atm_rows 1
atm_columns 50
textline asstabletxt1, 0,2
helpfile help/asstable.hlp
TEXTLIRE assesstat ("Print list of all assessments")
BUTTON BESONS (
     RUM_FORE 1
```

```
aum columns
       textline assesstat, 0 ,2
       helpfile help/mobelp.hlp
 TEXTLEME dbdstatzt ("Frist all database dates")
 BUTTON disdate (
      REEL COLUMNS
      textline dbdststxt, 0 ,2
helpfile help/dbdsts.hlp
 TEXTLINE dbheelpgtrt ("NAMENG: Actions you take on this screen affect
                            ASAN'S permanent detabases (*)
            @hsekpgaotion
            BUR TOWN
            am_colu
                           70
            1120
                           0,0, 0,77
            textline
                           dbheekpgtzt, 2,2
                           dodate,
                                        4,2, "CALL dumny"
6,2, "CALL pratabo"
            betton
            button
                           .....
                           asstable1, 8,2, "CALL SUpropt 1" asstable2, 10,2, "CALL SUpropt 2"
            button
            button
            button
                           updateinfo, 12,2, "CALL dumy"
       }
 SCHEEN dhisekpysoreen (
      title
                  "DATABASE BOUSEREEPING"
      Window
                  Chambergastica, 2,1
      window
                  majorantica5, 18,1
 WINDOW quickessbut {
        RUM COME
                       10
                       2
                mulbut00, 0, 0,
                                        "CALL SUprist deplault[0].arr"
        button
                malbut01, 1, 0,
                                        "CALL SOprint deplault[1].arr"
        betton
                  mmlbut02, 2, 0,
                                        "CALL SUprist deplacit[2].arr"
        button
                mulbut03, 3, 0,
                                        "CALL SUprint deplanit[3].arr"
        button
                 mulbut04,
                             4. 0.
                                        "CALL SUprist deplete[4].arr"
        button
                mulbut05, 5, 0,
                                        "CALL SUprist depletit[5].arr"
        bettoe
                malbut06, 6, 0,
                                        "CALL SUprist deplault[6] .arr"
        button
                 mulbut07, 7, 0,
                                        "CALL SUprist deplacit[7].err"
        button
                mmlbut00, 0, 0,
mmlbut09, 9, 0,
                                       "CALL SUprist deplanit[8] .arr"
        button
                                       "CALL SOprist depisult[9].arr"
         quickssebutl (
WINDOW
        num rows 10
num columns 2
        button mulbut10, 0, 0,
                                       "CALL SUprist depleult[10].arr"
        button malbut11, 1, 0,
                                       "CALL SUprint deplacat[11].arr"
                 mulbut12, 2, 0,
                                       "CALL SUprist deplault[12].arr"
        button
                 mulbut13, 3, 0,
                                       "CALL SUprist deplacit[13].arr"
        button
                 mulbut14,
                                       "CALL SUPrint deplanat [14] .arr"
        button
                 mulbut15, 5, 0,
                                       "CALL SUprist deplault[15].arr"
        betton
                 mulbut16, 6, 0,
                                       "CALL SOprint deplault[16].err"
        button
                malbut17, 7, 0,
mulbut18, 8, 0,
mulbut19, 9, 0,
                                       "CALL SUprist deplault[17].arr"
        button
                                       "CALL SUprist deplett[18].arr"
        button
                                       "CALL SUprist deplemit[19].arr"
TEXTLINE sussion ("Frint list of SUPERUSER's tables")
TEXTLINE descion ("Frint list of SEADQUARTERS' tables")
BUTTON
     AWA_rows 1
     ava_columns 50
     textline suselect, 0 ,2
helpfile help/nohelp.hlp
BUTTOM hqualect (
    num_rows 1
num_columns 50
     textline hquelect, 0 ,2
helpfile help/nohelp.hlp
```

```
WINDOW zetwopen (
   num_rows 1
num_columns 76
                  retassent, 0,1, "CALL mlista", "NEW_SCHOOL dhisekpgscreen"
WINDOW Agreement (
aum_rows 3
aum_columns 76
                  hquelect, 0,1, "CALL SUpropt 3" suselect, 2,1, "CALL SUpropt 4"
    betton
SCREEN SLORESCREEN {
    title "SELECT ASSESSMENT FOR DATABASE PRINTOUT"
                  hqsuselect, 4,1
receless, 8,1
quickssbut, 9,3
      window
      window
      WARROW
      window
                  quickessbut1, 9,42
                  ourdst, 9,5
cardstl, 9,44
      window
      window
                  retwopra, 20,2
      window
      border
                  X
          Declarations for view CERCE-LIST SCREEN
TEXTLINE otherchkist ("View another checklist")
NOTION othershilst (
      am_rows
      num_columns 25
      testiles
                      otherchkist, 0, 2
                   help/othrohk.hlp
      helpfile
TENTELOCIC fossi2 (
     filename txtblk/fonsi2.txt
num_rows 10
num_columns 76
      200_001
                      THE
      border
WINDOW foas12 {
                      13
     REST TOWN
     aum_columns 78
textblock for
                      foms12, 0,1
                    otherchklet, 12,2, "REMOVE WINDOW"
TEXTELOGE fossil (
fileness txtblk/fossil.txt
aum_rows 10
     awa_oolwas
                    76
     border
                      YES
WINDOW foasil {
     sam colamna
     textblock
                      fonsil, 0,1
                     othershklet, 12,2, "RIMOVE_WINDOW"
     button
TENTELOCK ceter3 {
    filenese triblk/ceter3.txt
    amm_rows 10
                   10
76
     aum_oolumas
     border
                     YES
```

```
WORDOW ontex3 (
      aun_rows 13
aun_columns 78
      tartblock cetar3, 0,1
betton otherchkist, 12,2, "REMOVE_WINDOW"
TEXTSLOCK cetax2 {
      filenene
                      txtblk/ostex2.txt
      PART LOAD
                     10
      awn_oolwan
                      76
                      774
WINDOW cater2 (
awa rows 13
     awa_rows 13
awa_columns 78
      textblock
                       ostezi,
                                 0,1
      bettoe
                     othershkist, 12,2, "REMOVE WINDOW"
THEFFELOCK cetazi (
     filenme tra
                      txtblk/cstex1.txt
     num columns 76
border 725
                      TES
WINDOW ceteri (
                      11
     Far Loan
      num_columns 78
      textblock
                      ceteri, 0,1
                     othershkist, 12,2, "REMOVE_WINDOW"
     bettoe
TRETLINE fonsiztrt (
"Show documentation necessary for finding of no significant impact")
MULTON foasi2 (
    num_rows 1
num_columns 72
     textline fomsi2txt, 0, 2
helpfile help/fomsi2.hlp
TEXTLEM fossilts (
"Show MEPA bases for finding of so significant impact (FOREI)")
MOTTOM fonsil (
    aum rows 1
     textline fonsiltxt, 0, 2
helpfile help/fonsil.hlp
TEXTLINE cataritat ("Show documentation needed for categorical exclusions")
303200 ostaz3 (
    awa_rows 1
     aum_columns 55
     tertline cetex3trt, 0, 2
helpfile help/cetex3.hlp
TEXTLEME ostes2txt (
"Show examples of proposed actions qualifying for categorical exclusions"}
MOTTON cetax2 (
    Rum rows 1
Rum columns 74
taxtline catex2txt, 0, 2
helpfile help/catex2.hlp
TEXTLIRE cetaritat ("Show NEWA bases for categorical exclusions (CATEX)")
BUTTON osteri (
   num rows 1
num columns 55
     textline cetaxitxt, 0, 2
helpfile help/cetaxi.hlp
```

```
weaklist
       num_rows 11
num_columns 78
                 osteri,
        bettos
                                   2,2, "ADD_WINDOW cetari 3 1"
                                   4,2, "ADD_WINDOW outar2 3 1"
4,2, "ADD_WINDOW outar3 3 1"
9,2, "ADD_WINDOW fonsil 3 1"
10,2, "ADD_WINDOW fonsil 3 1"
        bettoe
                   cater2,
       betton
                   ceters,
                   fonsil,
       button
       button
                   fond12,
SCREEN viewchecklistscreen (
     title
                 "VIEW CERCETIST"
     window
                 weaklist, 2,1
     window
                 majoractica4, 18,1
     berder
               774
SCREEN probetateureen (
  title
          "MIVIRORADITAL ASSESSMENT STATUS"
   mainscreen THE
  border
              YES
                             3,1
7,1
             assamoon,
   window
            probetet, 7,1
chgassess, 16,1
  Window
   window
              majorastica, 18,1
  Window
```

## A.3 Screen Description File for Report-Related Portions of ASAN

```
Declarations for REPORT GENERATION SCREEN
TEXTLES viewoisets: ("View text on calculated noise affects for current assessment")
BUTTOM Viewmoisetz (
    AUR FOWS
                1
    num_columns 72
    textline viewhoisetxt, 0 ,2
helpfile help/viewtext.hlp
TEXTLINE viewbplatetzt ("View boilerplate")
NUTTON Viewbplate (
    aum_columns 72
    taxtline viewbplstetxt, 0 ,2
helpfile help/rolodex.hlp
TEXTLINE pracisesbytzt ("Frint above text with associated boilesplate")
BUTTON pracisembp (
   REAL POWE 1
    ava columns 72
   taxtline prhoisembptxt, 0 ,2
helpfile help/rolodex.hlp
TEXTLINE precisetzt ("Frint above text")
BUTTOW prmoisetz (
    num_columns 72
   textline pracisetxt, 0 ,2
helpfile help/crestxt.hlp
```

```
MUTTON stdrapst (
     aum_columns 72
     textline stdreprtxt, 0 ,2
helpfile help/creetxt.hlp
TEXTLEM selectreptxt ("Actions you can now take to produce text and graphics:")
WINDOW reportantion (
          num_rows 15
num_columns 77
                         0, 1, 0, 76
selectreptxk, 4, 1
stdreprt, 6, 1, "CALL cellrpt"
viewnoisetx, 8, 1, "CALL dumny"
          1420
          testline
          button
          bettos
                        precisetx, 10, 1, "CALL callrpt3"
precisesbp, 12, 1, "CALL callrpt4"
viembplate, 14,1, "MEM_SCREEN bplaterevscreen"
          button
          Jecton
button
 )
SCHEEN reportgenscreen {
    title "MAKE A REPORT"
    mainscreen TES
     window reportantion, 2, 1
border YES
       · Declarations for boilerplate review screen
THETAINE otherbplatetat ("View other boilesplate text")
BUTTON otherbplate (
     FEET LOAD
      zwa_columns 30
      textline otherbplatetxt, o
helpfile help/blrplate.hlp
                          otherbplatetst, 0, 2
PERTHOUN heardon {
filesone blxplt/kringdeg.bpl
num_rows 10
num_columns 76
border YES
WEMDOW hearden (
     Far Lone
                         13
      aum_rows 13
mu_columns 78
taxtblook heardsm, 0,1
button otherbplate, 12,2, "RUMADVE_WINDOW"
TEXTLINE heardemants ("Hearing demage risk")
BUTTON hearden (
     aum_rows 1
aum_columns 35
textline heardemtxt, 0, 2
helpfile help/birplate.hlp
TENTALOCK structden (
      filename blrplt/strotdmg.bpl
      BAR LOAS
                         10
      Bun columns 76
      porder
                         724
WINDOW structdem (
     ON STREETHING 13
REM_FOUR 13
REM_columns 78
taxtblock structdem, 0,1
button otherbplate, 12,2, "REMOVE_WINDOW"
TENTINE structdentxt ("Structural demage")
```

TENTLINE stdreprint ("Frint stendard report")

```
NOTION structdom (
       RWG_FOW
        testine structdentri, 0, 2
helpfile help/hirpiste.hlp
  TEXTELOGE actistfor (
        filename birpit/sipintf.bpl
num_rows 10
num_columns 76
        border
                            YES
 WINDOW actintfer (
num_rows 13
num_columns 78
        taxblock sctintfer, 0,1
bettos otherbplate, 12,2, "REMOVE WINDOW"
       button
  TEXTLES actistferts: ("Activity interference")
  BUTTON actintfer (
     2m_row 1
nm_column 35
textline actintfertxt, 0, 2
helpfile help/blrplate.hlp
 TENTELOCK wildlife {
   filesame blxplt/wildlife.bpl
   aum_rows 10
   num_columns 76
   border TES
TEXTLUME wildlifetst ("Wildlife")
 BUTTOW wildlife (
      num_cow 1
num_columns 35
textline wildlifetxt, 0, 2
helpfile help/birplate.hlp
 TEXTSLOCK comintfer (
      fileame blrplt/spakintf.bpl
num_rows 10
num_columns 76
border YES
      border
WINDOW comintfor {
    NOW comistra-
aum_rows 13
aum_columns 78
textblock comistfer, 0,1
betton otherbplate, 12,2, "REMOVE_WINDOW"
TEXTLINE comintfertst {"Communication interference"}
SUTTON comintfor (
     aum_columns 15
textline comintfertrt, 0, 2
helpfile help/blrpiste.hlp
TEXTELOGE livestock (
      filename blrplt/livestek.bpl
aum_rows 10
aum_columns 76
     border
                         YES
```

```
WINDOW Livestock
      BUR FOW
      aum columns 78
      tarthlock livestock, 0,1
button otherbplate, 12,2, "REMOVE WINDOW"
      button
 TEXTLES livestocktat ("Economic demage to livestock")
 NUTTON Livestock (
      BWB_FORM
      awa_columns
                    35
      textline
                      livestocktat, 0, 2
      belpfile
                      help/blrplete.hlp
 FANT LOAM
      awa_columns
                      76
WINDOW manoy (
      num_rows 13
num_columns 78
                     annyace,
                                 0,1
                     otherbplate, 12,2, "NUMBOVE_WINDOW"
TENTETE annoyint ("Summa annoyance")
 BUTTON annoy (
     aum_rows
aum_columns
textline
                     35
                      ennoytet, 0, 2
      helpfile
                    help/blrplate.hlp
TEXTLINE bplatetat ("Select one of the following to view available
                          noise effects boilexplate:")
TEXTLINE returnpt ("Return to report selection screen")
BOTTOM returnspt (
   num columns 42
taxtline returnrpt, 0, 2
   helpfile help/nobelp.txt
WINDOW mulltest (
   ATEL_POWS 3
   mum_columns 77
   button returnept, 1, 1, "HEW_SCREEN reportgensureen"
WINDOW bplate {
     aws_oolumns
                   70
     1120
                    0,0, 0,77
     textline
                    bplatetat, 2,2
     button
                    аввоу,
                               4,2,
                                       "ADD_WINDOW amony 3 1"
     button
                    comintfor, 6,2,
                                         "ADD_WINDOW comintfor 3 1"
                    comintfor, e.z., "And "name" and interface 3 1"
hearden, 10,2, "AND WINDOW sotistfor 3 1"
livestock, 4,39, "AND WINDOW livestock 3 1"
wildlife, 6,39, "AND WINDOW wildlife 3 1"
struction, 8,39, "AND WINDOW struction 3 1"
     button
     button
     bettoe
     button
     button
SCREEN bplaterevecreen {
    title
window
                 "VIEW BOILERPLATE TEXT"
                bplate, 2,1
     Window
                 mulitest, 18, 1
     border
```

## A.4 Screen Description File for Graphic Portion of ASAN

```
-INCLUDE STATEMENT FOR THE ASAM & GRAPHICS typodef DEFINITIONS MEMBED TO
- COMPILE USUBS.C WITHOUT INCURRING THE WARTE OF THE COMPILER
INCLUDE ASSETTES.E
 INCLUDE ASAN.E
 - DECLARATIONS IN NON-LEXICAL OBJECT SEQUENCE ......
        Data structures for holding list of map names
 VARIANCE dlame[0] {
                   STR.DIG
                   4-04
  DATOM dlman0 {
    avariable dinem[0], 0, 0
pickable 20
VARIANIS dlass[1] {
type sTRING
  DATOM dlamm1 (
     ME TOWN
    variable dism[1], 0, 0 pidkable 20
VARIABLE dlamm[2] (
    type STRING
format 4-8s
  DATOM dlasm2 {
    BUR TOMS
    num_columns 8
variable dlnum[2], 0, 0
pickable NO
VARIABLE dlass [3] {
type string
  } temalb MOTAG
    num columns 8
   variable dlama[3], 0, 0
pickable 20
VARIABLE dlam[4] {
    type
format
  DATOM dlass4 (
    BUR_FOWS
    num_columns 6
   variable dinam[4], 0, 0
pickable 20
VARIABLE diama[5] (
    type
                STREET
                 4-8#
 DATEM dlams (
```

```
num_rows 1
num_columns 8
variable dinum[5], 0, 0
pickable NO
}
 VARIABLE diama[6] (
     type STRING
format 4-0s
 aum_rowe 1

Eum_columns 8

variable dlnem[6], 0, 0

piokable EO

}
   DATCH dlame (
VARIABLE dlama[7] {
     type STRING
format 4-8s
   }
DATOM dlamm7 {
     num rows 1
    variable dlass[7], 0, 0
pickable 20
VARIABLE dlasm[8] {
    type STRING
    formet 4-8s
   DATEM dlamas (
     aum_rows 1
    variable dlasm[8], 0, 0
pickable NO
VARIABLE diama[9] (
     type STRING
format 4-8s
   DATOM dinoms (
     hum_rows 1
    Num columns 8

Variabla dinum[9] , 0 , 0

pickable 20
VARIABLE dlamm[10] {
     type STRING
formet 4-8s
 aum_columns 0
variable dinam[10], 0, 0
pickable NO
}
  DATOM dlama10 {
VARIABLE dlamm[11] (
    type STRING
format 4-8s
  DATOM dlassii (
    aum_rows 1
aum_columns 8
variable dinem[11], 0, 0
pidable 20
VARIABLE dlamm[12] {
    type STRING
formet 4-8s
  DATOM dlnma12 {
    num rows 1
num columns 8
 _um_columns 0
variable dinam[12], 0, 0
pickable NO
}
```

```
VARIABLE dlamm[13] {
       type STRING
formet 4-8s
     DATEM diamets {
        num rows 1
        aum_columns 8
      Variable dinam[13], 0, 0
pickeble 20
  VARIABLE dlamm[14] {
       type STRING
formet 4-0s
     DATOM dlasmi4 {
      num_rows 1
num_columns 8
variable dlamm[14], 0, 0
pickable NO
               Declaration for entering coordinates
 VARIABLE estimt {
foundin "COORDIBATE est.lat"
type STRING
found %-13s
                                                                  - COMMENT or ENTER Set
 DATCM entlet (
ann rows 1
ann columns 25
        variable entist, 0 ,11
leader "Letitude: "
helpfile help/combmap.hlp
 VARIABLE estlong (
foundia "COORDINATE est.lon"
type STAIRG
format 4-13s
DATOM entlong (
num_rows 1
num_columns 25
       variable entlong, 0 ,11
leader "Longitude:"
helpfile help/combmap.hlp
      Declaration for show coordinates
VARIABLE shwist {
    foundin "COOMDIEATE show.lst"
    type STRIEG
    formet 4-13s
DATOM shwist (
      num_rows 1
num_columns 25
      variable shwist, 0 ,11
leader "Latitude: "
piokable BO
helpfile help/combmap.hlp
VARIABLE shwlong {
foundin "COORDINATE show.lon"
type STRING
format $-13s
}
```

```
variable skwlong, 0 ,11
leader "Longitude:"
pickable NO
helpfile help/combmnp.hlp
VARIABLE shedist {
     type DOUBLE
uplimit 10000.
lowlimit 0.
format $10.21f
DATOM shwdist (
      num_rows 1
num_columns 13
veriable shwdist, 0 , 0
pickable 20
trailer "km"
- DECLARATIONS FOR SCREEN READER
     foundin "ASAMERADR ASSESSMENT.nome"
    type STRING
format 4-30s
DATCM 'essessmel {
     ATE POWE
     aum_columns 78
      vaziable assessme, 0 , 30
     leeder
                     "Name of Gurrent assessment: "
     pickable
                    100
     helpfile help/mobelp.hlp
VARIABLE comment (
foundin "ASAMBADE ASSESSMENT.desc"
     type STRING
formet 4-66s
DATCM comment (
    num_row 1
num_row 1
num_columns 77
variable comment, 0 , 10
pickable 20
WINDOW lin (
     Par Loan
    atm columns 78
WINDOW assaumonn {
     aum rows 3
aum columns 76
     datum assessmel, 0,0 datum comment, 1, 1 line 2, 0, 2,77
- Declaration for buttons used in majoraction footer
TRITLINE probeta ( "REVIEW CURRENT ASSESSMENT STATUS" )
MUTTOW probetat {
     aum rows 1
aum columns 35
     textline probets, 0, 2
helpfile help/nohelp.hlp
TEXTLINE probetal ( "WORK ON AMOTEER ASSESSMENT" )
```

```
BOTTOM probetati (
      aum rows 1
aum columns 28
textiine
      textline probstsi, 0, 2
helpfile help/mobelp.hlp
  TEXTLINE probete2 ( "NEVIEW ASSESSMENT STATUS" )
  BUZZON probetat2 (
      num_rows 1
num_columns 26
      textline probsts2, 0, 2
helpfile help/nobelp.hlp
 TEXTLINE probdef ("ADD INFORMATION TO CURRENT ASSESSMENT" )
  MUTTOW probdef {
      BUR FOWS
      aum columns 39
     tertline probdef, 0, 2
helpfile help/probdef.hlp
 TEXTLINE probdef1 ("ADD TO ASSESSMENT DEFINITION" )
 BUTTON probdef1 {
     num_columns 31
     textline probdef1, 0, 2
helpfile help/probdef.hlp
 TENTLINE enelysis ( "AMALYEE DATA" )
 BOTTOM analysis {
      awa_oolwans 15
    textline analysis, 0, 2
helpfile help/detenel.hlp
 TEXTLINE reportes ( "MAKE A REPORT" )
 BUTTON report que {
     BAN LOAR
     awn_columns 16
     textline reportgen, 0, 2
helpfile help/reportgen.hlp
 TEXTLINE weekiles { "VIEW CHRONLIST FOR CURRENT ASSESSMENT" }
 MOTTON washklist (
     aum_columns 42
     textline vechklis, 0, 2
helpfile help/viewlist.hlp
TRATELINE detabaseing ( "MAKE DATABASE INQUIRIES" )
MUTTOW databaseing (
     RUM FOWS
    num_columns 26
textline databaseinq, 0, 1
helpfile help/makinq.hlp
         Declarations for majoraction footers
WINDOW majorastica (
                                -header for problem status screen
    num_columns 78
    1120
                     0, 0, 0,77
"Please select cae of the following actions:"
    title
                    probdef, 2, 0, "CALL dummy"
analysis, 2,43, "CALL dummy"
reportgen, 2,62, "CALL dummy"
    button
    button
    button
```

```
weaklist, 3, 0, "CALL dummy"
detabaseing, 3,52, "CALL dummy"
      betton
 WENDOW majoractical (
                                     -header for report generation screen
      Sem Tone
       aum_columns 78
       2120
                       0, 0, 0,77
       title
                        "Alternative sotions you can now take:"
                      probstaf1, 2, 0, "CALL dummy"
probstaf1, 2,29, "CALL dummy"
weaklist, 3, 0, "CALL dummy"
databaseing, 3,52, "CALL dummy"
       button
      button
      button
      button
             Declarations for MAP CONTROL
 VARIABLE shwarealtw (
        type IMMOUR
        format 4-84
uplimit 124
default 78
lowlimit 20
 DATOM shwareelDW {
       aum rows 1
        variable shwarealDM, 0 ,15
       leader "Show areas of "
trailer "(dB) <= LOW |
helpfile help/nohelp.hlp
                         "(d2) <= 122 <= "
 VARIABLE shwarealtwi (
       type INTEGER
       format 9-5d
uplimit 124
default 75
lowlimit 20
 DATOM shwareelDW1 (
      num_rows 1
num_columns 18
variable shwaresLDMI, 0 ,0
trailer "(dB)"
helpfile help/nohelp.hlp
VARIABLE shwarearsF (
      type INTEGER
format 4-5d
uplimit 124
default 75
lowlimit 20
DATEM shwaresPSF {
       aum_rows 1
num_columns 35
       variable shwaresPSF, 0 ,15
       leader "Show areas of "
trailer "(dB) <= PEF -
helpfile help/nobelp.hlp
                         "(db) <= PSF <="
VARIABLE shwareapsF1 {
      type DFFERR
format $-5d
uplimit 124
default 75
lowlimit 20
DATUM shwareaPST1 {
     RUE_TOWN 1
       variable shwaresPSF1, 0 ,0 trailer "(dB)"
```

```
helpfile help/sobelp.hlp
 TEXTLEME estimptofist ("Mark the map at:")
 TEXTLEM shwidptofint ("Locate touched point")
 BUTTOM shwidptofist (
      num rows 1
num columns 22
      textline shwidptofint, 0 ,2
helpfile help/sobelp.hlp
 TEXTLES shwdist ("Show distance between two touched points:")
 BUTTON shwdist {
      num_rows 1
num_columns 43
      textline shwdist, 0 ,2
helpfile help/mohelp.hlp
 TEXTLIRE addelmap {"Add or Remove layers being displayed"}
 NOTTON addalmap (
      aum_columns 77
      textline addelmap, 0, 2
helpfile help/mapsame.hlp
TEXTLINE addelmapinfo ("Add or Delete information on a map layer")
NOTICE addelmapinfo (
num_rows 1
num_columns 77
     textline addelmapinfo, 0, 2
helpfile help/addelmap.hlp
TEXTLINE showleseed ( "Show the legend" )
NOTION showleseed (
      aum rows 1
aum columns 17
textline showlegend, 0, 2
helpfile help/nohelp.hlp
TEXTLINE hidelegend { "Ride the legend" }
BUTTOW hidelegend (
   aum rows 1
aum columns 17
     textline hidelegend, 0, 2
helpfile help/nohelp.hlp
TEXTLINE erasedisplay { "Erase the map display" }
BUTTON arasedisplay (
     sum rows 1

sum columns 77

tartline erasedisplay, 0, 2

helpfile help/nobelp.hlp
TEXTLEME editoolors { "Edit the color assignments" }
MOTTON editoolors (
     aum rows 1
num columns 77
    textline editoolors, 0, 2
helpfile help/mobelp.hlp
VARIABLE layersevenes {
   type STRING
format 48s
```

```
DATEM layersevenese (
       num_rows 1
num_columns 77
       variable layerseveame, 0 ,35
leader "Save the displayed map as:"
helpfile help/savemap.hlp
 VARIABLE Vpeame (
type STRING
format 47s
 DAROM vpeame {
       awn columns
                            17
"View ("
       trailer
                               -):-
       verieble
       pickable
 TEXTLINE VM_G { "Solls.C" }
 207208 vw_a {
       sum_columns
      textline vw_c, 0, 2
 TEXTLINE TW_ma ( "Ajo.M" )
 SUTTON VV_NA (
ava_rows 1
      aum columns 7
      textline were, 0, 2
 FERTLINE Voms ( "Solls.M" )
SUTTON VW me {
    sum rows 1
    num columns 9
      tentline vw_me, 0, 2
TEXTLINE Vw_fa { "Ajo.F" }
BOTTON vw_fa (
aum_rows 1
aum_columns 7
textline vw_fa, 0, 2
TEXTLINE VV_fs { "Solls.?" }
BUTTON TW_fs (
      num_columns
    testline vw_fs, 0, 2
WINDOW mapetriaction (
               13
78
    NOT AND
    am_oolu
    detu
   datum estlat, 5, 21, "CALL lat2dec Seat", "NEWVALS"
datum entlong, 5, 48, "CALL lon2dec Seat", "NEWVALS"
button editoolors, 6, 1, "CALL edit colors"
button shwichtofint, 7, 1, "CALL show coords"
datum shwlat, 7, 25
```

The second secon

```
shwlong,
                                  7, 82
8, 1, "CALL show_dist"
                  shwdist,
      button
                  shwdist,
                                      0, 46
      detun
                 shwarealDST, 9, 3, "CALL dwmny"
shwarealDST, 10, 40, "CALL dwmny"
shwarealSST, 10, 40, "CALL dwmny"
shwarealSST, 10, 40, "CALL dwmny"
Layarsavename, 12, 1, "CALL store_screen"
                                      9, 3, CALL de
      detwa
      detun
      detw
      detwa
SCHEEN medecation (
    nains creen
                      YES
                                                                 ---- PARE PARE PARE PARE
       window assumnous, 2,1
window sepatrheation, 5,1
window sepatrheation3, 18,1
      Window
           Declaration for MAP MANDAMER
 DATOM GREVIAW (
       RWE TOWN
       awn_columns
                                22
       leeder
                                "CURRENT VIEW:"
       variable
                                 vpasma, 0 , 14
       pickable
 TEXTLINE layortst_tit { "AVAILABLE HAD LAYERS:" }
 TEXTRLOCK layertat {
     num rows 12
num columns 51
    border YES
Sileneme layers.txt
TEXTLINE layermow_tit ( "NOW DISPLAYED:" )
WINDOW layertst (
    am_rows 14
     ava_columns 51
     datum ourview, 0, 0
textline layertri_tit, 1, 0
textline layersov_tit, 0, 37
textblock layertri, 2, 0
WINDOW layarnow {
   aum_rows 17
   aum_columns 10
                   YES
     detun
                     dlmm0 , 1 , 1
                     dlam1 , 2 , 1
dlam2 , 3 , 1
     dete
     detu
                     dlnsm3 , 4 , 1
dlnsm4 , 5 , 1
     detm
                     dinem4 ,
     detus
      detun
                     dlnes5 , 6 , 1
                     dlnem6 , 7 , 1
dlnem7 , 8 , 1
dlnem8 , 9 , 1
dlnem8 , 10 , 1
     det
     detus
     detun
     datum
     detus
                     dlam10 , 11 , 1
                     disemi1 , 12 , 1
disemi2 , 13 , 1
     detu
     detun
     detu
                    dlmami3 , 14 , 1
dlmami4 , 15 , 1
     detw
VARIABLE layer2edd (
                ### (
###.THG
##
    type
     format
DATOM layer2add {
     num_rows 1
     num_columns 34
     leader "Add map layer"
trailer "to display"
```

detun

```
variable layer2edd , 0 , 15
  VARIABLE layer2del (
   type STRING
                     480
 DATOM layer2del {
    swa_rows 1
       swi_column 39
      leader "Remove map layer"
trailer "from display"
variable layer2del , 0 , 18
 WINDOW asklayer (
      ava rows 2
      aum_columns 41
      datum layer2add , 0 , 1 , "CALL add layer" datum layer2dal , 1 , 1 , "CALL dal_layer"
 TRATLINE donal ("Return to Map Control Screen")
  MOTTOW domai (
      num_columns 33
textline dome1, 0 ,2
helpfile help/newnsemst.hlp
 WINDOW | dome1 {
      num columns 50
button dome1, 0 ,2,
helpfile help/newssumt.hlp
                                           "MEN_SCHEEN mepcontrol"
 SCHEEN mapman ( title "NOW SCHEEN MONRAGMENT"
                    assnandom , 2 , 1
layertst , 5 , 2
layernow , 5 , 54
asklayer , 19 , 1
domal , 21 , 1
       window
       window
       Window
       window
       window
       border
 " Declarations for ADD OR DELETE INFORMATION FROM A MAP
 TEXTLES motavail ("Sorry, this facility is not yet available")
 WINDOW addelmapinfo {
      sum_rows 5
sum_solumns 78
textline notavail, 4,10
helpfile help/nohelp.hlp
 SCHEEN addalmaptafo (
      title
                    "ADD OR DELETE INFORMATION ON A MAP LAYER"
      window
                   assamoon, 2, 1
                 addelmapinfo, S, 1
domai, 14, 15
majoraction, 18,1
      window
       window
         Declaration for entering coordinates window
TEXTLEM entupplf
  ("Enter upper-left corner coordinates of area of current interest")
TEXTLEM entlowet
  {"Enter lower right corner coordinates of area of current interest"}
```

## Appendix B PROGRAM LISTINGS

The following listings are the C-language software modules of which ASAN is composed.

```
asan.pc -- ASAN Main Program
                1. Opens the printer.
                2. Calls strtASAN to establish communication with CRACIE
                3. Verifies that ASAN software is valid.
                4. Calls Uinit to start the sureen driver.
                5. At end of the session closes detabase and printer
 finaluda <stdio.b>
 finalude (process.b)
 #include <string.h>
 finelude (time.b)
 EDEC SOL REGIS DECLARS SECTION:
 EREC SQL INCLUDE faharris.h;
EXEC SQL INCLUDE hostwars.h;
EXCIC SOL END DECLARE SECTION:
 EXEC SQL INCLUDE SQLCA;
 Medine SQLCA STORAGE CLASS
 dinclude 'asaa.h'
main ()
 estern int MEN_DEROG_FENTURES;
int OC_u211_spheroid(), aloseORA(), logentry(), strtMSAH(), vfy_ASAH();
 void exposes ();
 static char *legal_notice[11] =
 ("\a\a\a\a\t\t\t RESTRICTED RIGHTS LEGEND\a\a",
 'ttree, deplication, or disclosure is subject to restrictions\n",
't as set forth in subdivision (b) (3) (ii) of the\n",
"\t as set forth in subdivision (b) (3) (ii) of the\n",
"\t Rights in Technical Data and Computer Software Clause\n",
"\t\t at 52.227-7013 of the DOD FAR Supplement.\n\n",
"\t\t\tBBW LABORATORIES INCOMPORATED\R",
"\t\t\t 10 MOUNTON STREET\n",
"\t\t\t 10 MOUNTON STREET\n",
"\t\t\t Combridge, MA 02238\n",
"\t\t\t 617-873-3000\n\n\n",
" User Interface Copyright (C) 1985, EMM Leborstories Incorporated\n",
"\t\t\t All Rights Reserved");
FILE *fopen();
pra = fopea("pra", "a");
if (pra — moul) (
    printf("\nCan't open printer!"); exit(128); }
dante = fopen("chronfil.mef", "a");
if (dente - NULL) (
    printf("\mCam't open chromfile("); exit(128); }
printf("ta[2J", 27);
for (i = 0; i < 11; i++) printf("%s", legal notice[i]);
printf("\n\n\n\n\t Fleese tap the space ber to continue, \"CTRL-C\" \
to abort. \a\a");
do {
    } while (1 != ' ');
printf("\nASAW starting....");
```

```
MEA_DESCS_FRATURES = 1;
 utmsone = 12;
 OC_m211_spheroid("clark66");
 strtagam();
 #1fdef CHRCHOTT
 printf("\asuccesfully connected as Project td (ts)\astrivileges are",
        ASSESSMENT.14, ASSESSMENT.2400);
 for (1 = 0; 1 < 3; 1++) printf(" to ", Assessment.auth[i]);
       You are now successfully COMMECTed to CRACIE. Compare the
 /*
       software actually running to what is in the validation file. */
 1f (vfy_ASAW()) exit(16);
 /*
       You are now successfully commercial to ASAM. Control is
       transferred to Uinit, whence it returns only at the very
         d unless something untoward happens along the way and
       MAN decides to pull the plug in midstream somewhere.
Uinit();
logentry();
 closeCRA();
printf("4a{24;1E",27);
 exit(0):
con2db.pd -- Routines connecting a VERSENAR to CRACLE, etc.
      This file contains:
     ASAFoosa
                   - connects any ASAN Assessment to CRACIE
      aloseORA
                  - closes ORACLE DEES
      lastsess
                   - retrieves last session from the logbook
     logestry
                   - makes an entry in the Assessment's logbook
     жуована
                  - fills in ORACLE user portion of ASSESSMENT structure
     Brollona
                  - earolls a new Assessm
                                         ent as an ORACLE User
      FUCCER
                  - connects ASAN'S SUPERUSER to ORACLE.
     ALAOTQ
                  - determines if a username exists in ORACLE.
**********************************
<d.usecorp ebrinait</pre>
                                   /* Header for calls to MS-DOS
finalede estáto.b>
Minima SQLCA STORAGE CLASS extern
                                /* Switch for header files */
/* All SQL declarations for this */
EXEC SQL REGIE DECLARE SECTION;
EXEC SQL INCLUDE hostwars.h;
                                      are in these two header files */
EXEC SQL INCLUDE faharris.h;
                                   /* this one comes from "U"
EXEC SQL ESD DECLARE SECTION:
EXTEC SQL INCLUDE SQLCA;
finclude "sean.h"
                                  /* Standard ASAN Sender File
ist Adamooss (sees)
ASAMoona -- Connect as ASAM Assessment to ORACLE
     Routine NOLLS BACK any outstanding transactions of the current
      user and then connects to the requested user. Returns an error
      code which indicates whether or not the COMMECT was successful,
    Notes: 1. The existence of the assessment on the detabase is
                assumed. ( vfyOid(name) is available to check.)
             2. If the name requested is not a "real" assessme
                but one of the privileged ORACLE names (used for
                system maintenance, DEA functions, etc.) the COMMECT
                will fail (sqlma.sqloods - SQL MAD LOGOS) . If so,
                you will be re-connected to the old assessment and
                the error code of the failed connect is returned
                to the calling program.
```

```
register int 1, j;
 ist temp, logestry(), loggedon(), myCasme(), recon, SUconn();
 Told empage():
 cher *clock();
 #1fdef CHBCROOT
   printf("\mASAMoom: %s", memo);
 Seedi C
loggedon();
                      /* Verify that we know who the person responsible is */
 SLOUT("Please stand by: Switching assessments....");
logentry();
                     /* Make entry in log book and disconnect previous user */
 WEEREAMI (Screen, Window, Datum, Sutton);
missiabl.arr[0] = misdesc.arr[0] = '\0';
 misslabl.leg
                 = misdesc.lem = 0;
 if ( (sqlos.sqloods -= 0) ||
                                           /* These are reasonable return codes */
      (sqlos.sqloods - NOT LOGGED ON) ||
       (sqlos.sqloods - NOT_COMMECTED) )
    stropy ( uid.arr, name ); /* Note: The SQL COMMECT statement */
uid.lem = strlem ( uid.arr ); /* doesn't like an immediate password */
stropy ( pwd.arr, univpwd); /* and a dynamic username.... */
    stropy ( pwd.arr, univped); /* and
pwd.len = strlen ( pwd.arr );
EXEC SQL COMMET : uid IDENTIFIED BY :pwd;
    temp = (int) sqlca.sqlcoda;
    Bifdet CHRCHOTT
    printf(" Connect = %ld", sqlca.sqlcode);
    -
    switch(temp) (
                                               /* What did ORACLE come up with? */
    -
                                               /* COMMECT Succeeded as planned */
       fprintf (dente,
                                               /* Load the ASSESSMENT Structure */
      "\nes Connected es to ID = ed (es)", /* and if that was o.k. you can */
         slock(), plannam.arr,
                                              /* Fick up the description from */
         Mermo, wid.err);
                                              /* the site's table of contents */
       if (lmyOneme()) {
          MIGG SQL SELECT description
               FROM table of contents
               INTO :workspace
                WEERE identifier = : Merro;
          #1fdef CEECEOUT
            printf(" %ld", sqlos.sqloode);
          Bearle #
          if (!sqlca.sqlcode) { /* Found a description in table of contents */
             workspace.arr[workspace.lem] = '\0'; /* Truncate description */
j = (sizeof ASSISMENT.desc) -1; /* to fit on the screen */
             if (workspace.lea >= j) (
                for(i = 0; i < j; i++) ASSESSMENT.desc[i] = workspace.arr[i];
                ASSESSMENT.desc[j] = '\0'; }
             alse stropy(Assessment.desc, workspace.arr);
          alse ( /* No Description.... Now can this beffff */
             fprintf(dante, "\n*s ASAWoonn: %s not in T.O.C", clock(), wid.arr);
fprintf(dante, "\n\t\t\t %s", sqlcs.sqlerm.sqlermc);
             SLOUTER ("Security Violation: Table of contents error");
/* This will become a security violation!
             GloseORA();
             exit (255) ;
                               But for now, let it go by */
             stropy (ASSESSMENT . desc,
                     "Description missing from table of contents");
            3
         EXEC SQL SELECT TO_CHAR(SYSDATE, 'DD-Mon-YY EE24:MX:SS')
         FROM dual INTO :timest1;
         printf("%ld", sqloa.sqloode);
         Sendi f
         EXEC SQL SELECT TO_CEAR(start_work, 'dd-Mon-yy EE24:MX:SS')
                   FROM logbook
DFTO :startdate
                   WEEKE start_work =
                           (SELECT MIN(start_work) from Logbook);
```

```
#1.fdef CHICKOUT
                     printf("%ld", sqlom.sqloods);
                    if (sqlos.sqloods) {
                          startdate.len = timest1.len;
                                 stracpy(lastdata.arr, timestl.arr, timestl.len);
lastdate.len = timestl.len;
                                 lastatr.lea = 0;
                                lastmon.len = 0;}
                          alse (
                                EMPONING -
                                Sprints(dante, "\nts ASANconn: ts logbook error",
                               clock(), wid.arr);
sprintf(daxte, "\n\t\t\t es", sqlos.sqlorms.sqlorms); }
                   alea (
                         MING SQL SELECT lastmir, lastmon
                                           FROM Logbook
                                            1970 :lastmir, :lastmon
                                            WHERE stop_work =
                                                         (SELECT MAX(stop_work) from logbook);
                         #1fdef CENCEDOT
                        printf("%ld", sqlom.sqloods);
                         feedif
                  lastatr.arr[lastatr.lea] = '\0';
                  lastmon.arr[lastmon.lan] = '\0';
                 /* All that's laft now is to put up the screen */
                 MEW_SCREEN("probetatecreen");
                 MENVALS ();
                 if (ASSESSMENT.auth[2] 1= 'T')
                       SLOUTES ("This assessment does not have resource authorization");
                 return (int) 0;
          also (
                Electronic -
                 Sprintf(dante, "\ats Mysterious failure on ts", alock(), mid.arr);
                fprintf(dante, "\n\t\t\t %s", sqlos.sqlarm.sqlarms);
fprintf(dante, "\n\t\t\t while retrieving info from SYS.VARNOUSER");
                temp = (int) sqlca.sqlcode;
               SHOOTER ("Undiagnosible system failure. ASAN will restart");
stropy (ASSESSMORT.neme, "SUPERGER");
               NEW_SCREEN ("firstscreen");
               breek: 1
   /* Trouble in River City if you get this far down in the switch .... */
  CRES SQL_NO_TERRORISE:
        SLOUTER ("You cannot use a blank as a name");
  DESCRIPTION OF THE PARTY OF THE
  Case NOT LOGOND OFF:
        ENDOMES:
        fprintf(dante, "\nts Authorization failure on te", clock(), uid.arr);
fprintf(dante, "\n\t\t\t te", sqlca.sqlarms.sqlarms);
sprintf( workspace.arr, "Unsuccessful: te",
                          sqlos.sqlerm.sqlermed);
       SLOUTER (workspace.arr);
       breek;
 default:
      THE COURSE
       fprintf( dants, "\nts Unsuccessful logon: ts", clock(), wid.arr);
       fpristf( dante, "\a\t\t\t ", sqlca.sqlerms.sqlerms);
       sprintf(workspace.arr, "Vasuccessful: %s", sqlca.sqlerm.sqlermac);
      ELOUTES (WOTESpace.arr);
) /* Bad SWITCE */
stropy ( wid.arr, Assessment.name);
                                                                                    /* Re-COMMECT with */
wid.len = strlen ( wid.arr );
                                                                                   /* previous name.
if ( stromp("sorenoser", wid.arr) ) (
     EXEC SOL COMMET : wid IDENTIFIED BY :pwd;
      recom = (int) sqlcm.sqlccdm;
     /* Re-initialize the screen. Logentry() call has closed all cursors: */
if (!recon && (!stromp(Screen, "chgourasscreen")) ) {
          pechgase();
```

```
return temp; }
    else recon = #Tooms();
    1f (recon) {
                          /* Now we're stuck beyond help */
      SLOUTEF ("ASAW got \"stuck\" looking for the data....");
      fprintf( dante, "\nts Adamoun stuck: ts", aloak(),
               sqloa.sqlerm.sqlerme);
      fprintf( dexts, "\m\t\t\t After strespted re-connect to %s", mid.arr);
      aloseORA();
      exit (255);}
    SLOUTED ("ASAN must restart due to an internal error");
   MEN SCREEN ("firstscreen");
   return temp;
 alse { emptmsg();
                                  /* If you get here, you've been doing */
    fprintf (dente.
                                  /* something outstandingly bisarre a */
    "\nts Forced shut-down by ts", /* long time before this function was */
                                  /* called. We idea how to recover! */
    aloak(),
   sqlos.sqlerms.sqlerms);
    GloseORA();
   exit (255) ;
}
 aloseORA()
 closeCRA -- Discounset as ASAN Assessment from CRACLE
                          and return ORACLE to original state as
                          given by 00_roods.
       Routine ROLLS RACK any outstanding transactions of the current
       user and then disconnects from ORACLE. Returns an error code
       which indicates whether or not the call was successful.
     Hotes: 1. The value of 00 code is set in the main() function
as the return code of the initial logon attempt. If
these codes change in subsequent releases of CRACLE
                   this function must be recompiled with the new codes.
               2. Unless the current user is SUPERCEER or this function
                   is called because of an unrecoverable ORACLE error,
                   calls to closeCRA() should be preceded by logestry();
int doods, falose(), spawnlp();
char *clock();
fifdef CEBCEDOT
printf("\adloseCRA");
Access of
SLOUT!"
                               *** End ASAH ***");
fprintf(dante, "\nts ASAN Close-down process:", slock());
EXEC SQL ROLLBACK WORK RELEASE;
Middel CERCEOUT
printf(" rolled back %ld", sqlca.sqlcode);
-
1f (00_reeds - 0) {
                        /* We're done if ORACLE was up when we started */
   fprintf(dante, " Servers retained intact");
if ((00_reeds == ORA_UMRATLABLE) || (00_reeds == -3120)) {
   doods = spenalp() Wall, "ior.ess", "ior.ess", "shut", NULL);
   #1fdef CEECHOUT
   printf(" 4d", doode);
   -
   fprintf(dante, "\ats Shut down ORACLE Server (%d)", clock(), doods);
   if (00_roods i= ORA_UMAVAILABLE) (
      doods = spawnip (P_WALT, "remors.ame", "remors.ame", "all", NULL);
     #1fdef CENCROOT
     printf(" %d", doods);
      Bee-11 7
     fprintf(dante, "\nts Deinstelled SQLFME (td)", clock(), doods);
     return doode;
```

```
else { /* Could be ASAH wasn't installed, or .... */
fprintf(dante, * Servers left machanged (Code %d)*, CO_roode);
 fprintf(dexte, "\n");
 falose (dazte) ;
 falose (pra) ;
 /********************************
                     Retrieves last entry from the ASAN logbook. It is
                       used when we need to establish the last activity on
                       the system. (SUFERUSER does not log its activity.)
       Returns the value of sqloa.sqloods to calling program
 fifter concentr
 printf("\mlasteess ");
 Bends f
 SLOUP("Re-loading the last session's parameters");
 EXEC SQL SELECT planar, To_CEAR(stop_work,'DD-Mon-YY EE24:MI:SS'), idnumber
         FROM: lastlog */
          FROM last_logia
          INTO :plantist, :lastdete, :userno; -
 #1fdef CHRCHOUT
 printf("%ld", sqloa.sqloods);
 Sendif.
if (!sqloa.sqloods) {
                                    /* This is what happens when all is 0.K. */
   planrist.arr[planrist.lea] = '\0';
    lastdate.arr[lastdate.les] = '\0';
else if ( sqlom.sqloods == SQL_NOT ) { /* When you start the very first time
   stropy(planrist.arr, "Wo work ever done yet");
planrist.arr[21] = '\0';
planrist.len = 21;
lastdate.arr[0] = '\0';
   lastdete.les
alse {
   fprintf (dente,
     "\nts Incomprehensible return code tld retrieving last session",
   clock(), sqlca.sqlcode);
fpristf(dante, "/n/t/t/t %s", sqlca.sqlcrrm.sqlcrrmc);
   emponeg () ;
#1fdef CENCEDUT
printf("\nts, ts, td", plantlet.arr, lastdate.arr, userno);
retura (int) sqlos.sqloods;
ist logestry()
           Makes an entry in the ASAW register for the work just
performed on the assessment. It is called by the
      logestry --
                      ASAMoonn and SUconn routines when they close out the
                     previous assessment.
      Neturns the value of sqlca.sqlcode to calling program
```

```
int roods;
 cher *clock();
 Bifdef CERCEDOT
    printf("\alogentry: ");
 if (strong(ASSESSMERT.neme, "SUFERUSER")) { /* Make entry in log book if the */
                                       /* previous same was not SUPERUSER. */
/* Terminate whatever is outstanding */
    EXEC SQL ROLLBACK WORK;
      fdef CHECKOUT /* One normally expects 0 here, but */
printf(" *ld", sqloa.sqloods); /* occasionally scattling horrid has */
    fifdef CENCROUT
      2116
                                       /* been known to pop up here.
    if (sqlos.sqloods != 0) {
       EMPCANG:
       Af ((eqlos.eqloods |= NOT_LOGGED_ON) &&
           (sqloa.sqlooda != NOT_COMMECTED)) {
          sprintf(workspace.arr, "Logbook choked on: %s",
                  sqloa. sqlarma.sqlarma);
          fprintf(dante, "\nes #s\t\t\t making logbook entry for #s",
                  clock(), sqlca.sqlerm.sqlermc, Assessment.ame);
          SLOUTER (workspace.arr);}
    alse (
             /* All is well and we are ready to make the logbook entry
         erno = ASSESMENT.14;
       EXEC SQL SELECT TO_CHAR(SYSDATE, 'dd-mon-yy EE24:MI:SS')
               FROM DUAL INTO :timest2;
       BING SQL MESERT MFTO register (pleaser,
                                                 lastatr, lastaca.
                                    start_work, stop_work, idamber)
                      VALUES (:planrasm, :lastmir, :lastmon,
                              TO_DATE(:timest1,'dd-mon-yy EE24:MI:SS'),
                              TO DATE(:timest2,'dd-mon-yy EE24:MI:SS'),
                              : TSGIBO) ;
       #1fdef CENCROUT
         printf("IMSERT %ld", sqlca.sqlcode);
         كناه
      if (roods = sqlos.sqloods) ( /* Fast in case it fails....
          fprintf(dante, "\ats Logbook entry failed: %s",
                     clock(), sqlca.sqlarm.sqlarmc);
         planram.arr, timestl.arr, timest2.arr);
EXEC SQL ROLLBACK WORK RELEASE;
         #ifdef CENCEOUT
            printf(" Rolled back *ld", sqlos.sqloods);
         Bearing C
      else {
         EXEC SQL COMMET WORK RELEASE;
         #1fdef CEECEOUT
           printf(" Committed %ld", sqlom.sqloods);
         Sendif.
         fprintf(daste, "\ats Disconnected to from ID = td (to)",
                     clock(), plantnes.arr, userno, Assessment.asse);
      return roode;
alse {
                                     /* If you are SUPERUSER, just make */
   EXEC SQL ROLLBACK WORK RELEASE; /* a call to log off and return
   #1fdef CHBCEOUT
     printf(" %ld (SU Bypassed)", sqlom.sqloode);
   Bearing #
return (int) sqlog.sqloods;
ist myOsmo()
/********************************
                    To set the ORACLE USER information in the ASSESSMENT
                    structure so it is available to all ASAN routines.
                    Looks in view SYS. VARISUSER and retrieves USERID,
                    MYERME, TIMESTANG and authorization codes.
     Returns the value of sqlcm.sqlcode to calling progrem
********************
```

```
#1fdef CENCHOUT
printf("\myOneme: ");
-
feeds !
EXEC SQL SELECT myid, myprive, mynemo
               FROM sys. viespuser
               METO : MARINO, :O suth, :my name;
my_nume.arr[my_neme.len] = '\0';
#1fdef CRECEOUS
printf(" tld - td (te) \x", sqlox.sqloods, my_neme.len, my_neme.arm);
-
lead1£
                                          /* Store results in global */
stropy(ASSESMONT.nemo, my_nemo.err);
                                          /* ASSESSMENT structure
ASSESSMENT .1d = Mearno:
ASSESSMENT.auth[0] = 0_auth.arr[0];
ASSESSMENT. auth[1] = 0_auth.arr[1];
ASSESSMENT.auth[2] = 0_auth.arr[2];
return (int) sqlos.sqloods;
ist prolices(sees)
arolloga -- ASAN'S SUPERUSER Grants COMMECT and RESCURCE
                   to new ORACLE user
     Noutine a). Signs on as SUPERUSER, if current user not DEA.
               b). Enrolls a new ORACLE USERSME.
              d). Signs on under the new assessment's name.
   Returns to calling program with sqlom.sqloods of last step amounted.
Boutine terminates immediately if a sub-step returns a non-zero code.
      Returns (EERO) if successful or (NOW-EERO) if not successful.
             Most likely, in order of appearance
             -0001 - Home already exists
              -0954 - No IDENTIFIED BY clause (i.e. blanks in name)
              -0987 - Illegal character in usernes
              -anna - Any other CRACLE error that can occur
char ame[];
int dlen, roods, wfyOid(), #Uccan();
register ist i, j;
cher *clock();
static char imputline[] = "emtdesc ";
printf("\narollORA");
وعلمة
SLOUT("A moment, please, while ASAM updates its detabase.....");
Bearis f
if ( roods = 5Uccen() ) { /* SUPERUSER's services are retired to do this */
   #15def CEBCHOOT
    printf(" %ld", sqloa.sqlooda);
   Beadif
   sprintf(workspace.arr, "SUPERUSER LOCKE REQUEST CRASSED Error %d.",
          roode);
   SLOUTEF (workspace.arr);
   SLOUIF ("You need serious help!.....");
   fprintf(dante, "\nts MROLLORA: Superuser logon crashed", clock());
   EEDOMS:
   Sprintf(dexte, "\z\t\t\t 4s", sqlos.sqlerms.sqlerms);
   aloseORA();
   ezit (255) ;
                                 /* Convert name to uppercase because */
j = n2bv.len = strlen(neme);
                                  /* ORACLE keeps USERGOME in uppercess */
for (1 = 0; 1 < j; 1++)
  n2bv.arr[i] = name[i] = toupper(name[i]);
n2bv.arr[n2bv.len] = '\0';
stropy(workspace.arr, "GRANT COMMECT, RESCURCE TO "); /* The GRANT statement */
struct (workspace.arr, name);
                                                   /* ment is not part of */
stroat (workspace.arr, " IDENTIFIED BY ");
                                                   /* AMSI standard SQL! */
stroat (workspace.arr, wmiwpwd);
workspace.lea = strlen(workspace.arr);
```

```
EREC SQL EXECUTE DAMBIATE :workspace;
Bifdef CHRCHOTZ
printf("\nts %ld", workspace.arr, sqloa.sqloods);
-
if (sqlos.sqloods) { /* Who knows what evil lurks in the heart of CRACLE? */
   fprintf(deste, "\ats Failed to earol %s/a/t/t %s",
                  clock(), name, sqlom.sqlerzmc);
   return (int) sqlom.sqloode;) /* This traps any illegal characters */
                                   /* and other strange things....
fprintf(dexte, "\nts Successfully enrolled ts", clock(), name);
EXEC SQL SELECT userid
               FROM systematist
               IMPO : ELd
               WHERE userness = :n2bv;
                                  /* User may enter up to 240 */
/* characters of drivel to */
/* identify this assessment */
ADD_WINDOW( "newsdesd" , 5, 5);
workspace.arr[0] = '\0';
for (1 = 0; 1 < 4; 1++) {
   imputlime[7] = i+49;
   UPDATE DATOM (impetline);
   entdesc[i].len = strlen(entdesc[i].arr);
   if (entdess[i].les - 0) break;
   street (workspace.arr, estdess[1].arr);
   if (entdesc[i].len < 50) break;
workspace.len = strlen(workspace.arr);
REMOVE_WINDOW();
EDEC SQL DESERT DETO table_of_contents (idnumber, description) VALUES (:wid, :workspace);
if (sqlos.sqloods) {
   fprintf(dexte, "\m** Table of contents entry failed\n\t\t\t *s ",
                aloak(), sqlca.sqlerm.sqlerma);
   SLOUTP (sqlos.sqlorm.sqlormo);
   roods = (int) sqlos.sqloods;
                                     /* This will cause the assessment */
   EXEC SQL ROLLBACK WORK;
                                     /* to become invisible to ASAN or */
   #1fdef CHRCHOUT
                                    /* may lead to security violation */
   printf(" %ld", sqloa.sqlooda);
                                     /* errors that will stop execution */
   .
Bendië
alsa (
   EXEC SQL COMMETT WORK;
   roode = ASAHoona (name) ;
                                     /* Once you get this far, sign on. */
   #1fdef CEECROUT
                                     /* This better never be mon-sero! */
   printf(" %ld", sqloa.sqloods);
   Seedif
   }
return roode:
int SUccess()
Connect ASAN'S SUPERUSER to ORACLE
        Returns SQLCA.SQLCODE
static char deaud[] = {0123,0125,0120,0105,0122,0125,0123,0105,0122,057,
                       0115,0105,0120,0110,0111,0123,0124,0117,0120,0110,
                       0105,0114,0105,0123,'\0'};
int temp, logentry(), myOnome();
#ifdef CENCROUT
printf("\astocea: ");
Bendif
logestry();
             /* Make logbook entry and terminate whatever is outstanding */
stropy( mid.arr, dbemid );
                                   /* Conmect */
uid.len = strlen( uid.arr );
EXEC SQL COMMENCE : wid;
```

```
#1fdef CENCEOUS
 printf(" %ld", sqloa.sqloods);
 -
 if (!sqloa.sqloods) myCamma();
stropy(ASSESSMEET.desc,"Owner of ASSE's Administrative Information");
 return (int) sqlom.sqloode;
 ist vfyOid(same)
 vfyOid -- Verify that a USERGINE exists in ORACLE's Dictionary
     Determines if the username exists from the SYSUSDAIST View.
      Returns SQLCA.SQLCODE of the SELECT statement.
 char assoil:
                      /* Argument is pointer to character string */
                      /* containing the username to validate
 register ist 1:
#1fdef CENCEOUT
printf("\mvfy0id");
 -
n2bv.len = strlen(name); /* Set up for WHERE dlause of the SHLECT */
for (i = 0; i < n2bv.len; i++) /* CRACLE likes uppercase names */
n2bv.arr[i] = toupper(name[i]);
EDEC SQL SELECT userid
        FROM systerlist
NFTO :userno
        WEERE tearness = :a2bv ;
#1fdef CERCEOUT
    printf(" %ld", sqloa.sqloods);
return (int) sqlom.sqloode;
housempt.pc -- Set of routines to print "housekeeping" data from
                  the ORACLE Data Dictionary.
    Routines in this file:
      #Oprint
                - Allows SUPERUSER to print an assessments tables
                - Selects level of detail for SUprint or prints
      SOpropt
                   BEADQUARTERS or SUPERUSER tables with that option
                - Frints names and comments on tables for assessment
     pratool()
                - Frints names of all tables and names and comments
                   of columns within tables
      pratebo() - Prints TABLE_OF_CONTRETS entries
***********************************
#include <stdio.b>
                              /* The usual stuff, of course */
/* Beader for calls to MS-DOS */
d. second oprocess.
#include <string.h>
                             /* String manipulation header */
#define SQLCA_STORAGE_CLASS extern
EXEC SQL INCLUDE SQLCA;
                             /* SQL Communication Area
EXEC SOL BEGIN DECLARS SECTION:
EXEC SOL INCLUDE hostware.h:
RICEC SQL END DECLARE SECTION;
#include "esentype.h"
finciude "asan.h"
                            /* Standard ASAW Booder file */
static printoption = 0;
ist SUprist (name)
```

```
SOprist ()
                --- Frint tables for an assessment (called from the
                       housekeeping sureen while SUPERUSER is connected
      Nowtine logs on as the named assessment, prints tables as that
      assessment, returns to SUPERUSER, puts up housekeeping screen again.
char asso [];
char *clock();
stropy (wid.arr, asso);
wid.lea = strlea(wid.arr);
stropy (pwd.arr, waivpwd);
pwd.lea = strlea(pwd.arr);
MINEC SOL BOLLBACK WORK RELEASE;
MANG SQL COMMICT : wid IDENTIFIED BY :ped;
if (sqloa.sqloods) (
   EXPONENT:
   fprintf(deste, "\nes Could not connect to to for printing tables",
   clock(), name);
Sprintf(dante, "\n\t\t\t %s", sqlos.sqlerm.sqlerme);
   SLOUTEP ("Could not connect to assessment");
   SLOUIP (sqlos.sqlerm.sqlermo);
ales {
   myceme ();
   if (printoption == 2) pratool();
else pratabs();
stropy(ASSESMENT.mems, "SUPERUSER");
5000aa () ;
if (sqlca.sqlcode) (
   fprintf(dante, "\n4s Could not reconnect after printing tables for 4s",
   glock(), asse);
fprintf(dante, "\n\t\t\t %s", sqlca.sqlarma;;
slcorms("Could not reconnect to SUPERUSER");
   SLOUTS (sqlos.sqlorm.sqlorms);
   exit (255);
MRN_SCREET ("dbkselpgscreen");
return (int) 0;
int SUpropt (value)
SUpropt() --- Set options for printing assessments or print the
                      SEADQUARTERS or SUPERUSER tables
     Noutine logs on as the named assessment, prints tables as that
     assessment, returns to SUPERUSER, puts up housekeeping screen again.
int value:
cher *clock();
#11def CENCEOUS
  printf("\msopropt %d", value);
Seedif.
switch (value) {
case 1:
  printoption = 1;
  if (mlisto()) {
     BEDOMES:
     SLOUTS (sqlos.sqlarum.sqlarume);
     return (int) -1;}
  blakdspl();
  MEN_SCREEN("sloasscreen");
  Aparet () :
  return (int) 0;
0856 2:
  printoption = 2;
  if (mlisto()) (
     EEEDCares:
     SLOUTS (sqlca.sqlerm.sqlerme);
```

```
return (int) -1;}
    blakdepl();
    MER_SCREEN("slongscreen");
    ubunch () ;
    return (int) 0;
 case 3:
 wliste();
 optr1 = hquoan;
 EXEC SQL ROLLBACK WORK RELEASE;
 EXEC SQL COMMET : opt:1;
 if (sqlos.sqloods) {
    EDOMG:
    fprintf(dasts, "\ate Could not connect to MEADQUARTERS for printing tables",
    clock());
fprintf(dante, "\a\t\t\t %s", sqlca.sqlcrrm.sqlcrrmc);
sLCUTEP("Could not connect HEADQUARTERS");
    SLOUTS (sqloa.sqlarm.sqlarma);
 alsa {
    myoname ();
    if (printoption -- 2) pratool();
    else pratabs();
 stropy(Assisment.nems, "SUPERCEER");
 STOORS ():
 if (sqloa.sqloods) {
    ENDOMES;
    fprintf(deste, "\nts Could not reconnect after printing MEADQUARTERS tables",
                     alock());
    fprintf(dexte, *\a\t\t\t *s*, sqlca.sqlerma.sqlerma);
    SLOUTER ("Could not reconnect to SUPERUSER");
    SLOUTS (sqloa.sqlerma.sqlerma);
    exit (255) ;
NEW_SCREEN ("dbheekpgscreen");
return (int) 0:
 default:
wlisto();
if (printoption - 2) pratool();
else pratebs();
HEW_SCHEEM ("dbhselpgscreen");
return (int) 0;
int pratabe()
pretabe() --- Print a list of all tables for this assess
      Routine opens cursor U2 then fetches rows until SQL_BOF is found. Each row printed on the system printer. Then the cursor is closed.
      Hote: Modifications to this function may impact the related
functions tlisto(), tlistf() and tlists() that open the
sursor, fetch rows using it respectively, and close it
**********************************
Sdefine Pagester 54
Tunded COMMENTERACE
#define COMMETERACE 54
int i, j, line, lop, mop, page;
                                                /# Counters for lines, etc.
int tlisto():
                                                /* Detabase Utilities Used
ist tlistf():
int tlisto();
SLOUT("Frinting tables");
if ( | tlisto() ) {
                                              /* Open sys.viezptab query
  j = 0;
   page = 1;
   fprintf( prm, "\t\tInventory of Tables for %s\t\tPage %d\n",
            ASSESSMENT.name, page);
  while ( sqlcs.sqlcode != SQL_NOF ) (
```

```
/* Get zame of the next table */
        15 ( line >= PAGESTEE ) (
                                                /* Make sure that you have space */
           Sprintf( pra, "\f\t\tImvestory of Tables for %s\t\tPage %d\a",
                    ASSESSMENT.zame, +ipage);
        j++; /* Frint name of the table */
Sprintf(prn, "\n44d 4-20s ", j, tid.arr); /* Assumes <= 20 char! */
       lop = 0; /* Frint as meny characters as the comment field contains */
        while (lop < workspace.lea) {
           map = (lap+communityracz < workspace.lam) ?
          for ( ; acp > lcp; acp--) /* Find good spot to end this line */
if (workspace.arr[acp] == '\0') break;
          if (lop == nop) { /* In case we have a line without white space */
fprintf(prn, " ");
              for ( ; acp < lop+communityAcm; acp++)
                 fprintf(prm, "%a", workspace.arr[acp]);
fprintf(prm, "\a\t\t\t ");
                 lime++; }
           also { /* The normal case when we can find a white space somewhere */
              workspace.arr[ncp] = '\0';

Sprintf(prm, "%s\m\t\t\t", &workspace.arr[lcp]);
              lize++;
             lop = acc+1; }
          lineit:
       ) /* End WHILE (Lop)*/
). /* End WHILE (sqlos) */
    fpriatf(pra, "\a\a\t\t\t\t= == -\f");
   tlista();
#1fdef CHBCROTZ
 else {
   printf("\a\a00021 My oursor did not open!");
    exponeg();
int pratcol()
pratcol() --- Print a list of all columns CREATED BY this
                          assessment sorted by table in which they occur
      Routine opens cursor V4 then fetches rows until SQL BOF is found.
Each row printed on the system printer. Then the cursor is closed.
      Note: Modifications to this function may impact the related
functions talisto(), talists() and talists() that open the
sursor, fetch rows using it respectively, and alose it
Sdefine PAGESTER 54
funded COMMERCEDACE
Marine Consummer 40
int talisto();
                                                /* Detabase Utilities Used */
int tolistf();
int tolisto();
ist i, j, k, line, lop, mop, page;
                                                /* Counters for lines, etc.
char qual_col[31];
SLOUT("Printing tables and columns");
12 ( | telisto() ) (
                                                /* Open "COL" Detablot Query
  1 = k = 0;
   page = 1;
   Sprintf( prn, "\tlaventory of Columns by Table for ts\t\tPage td",
           Assessment name, page);
   11me = 3;
  stropy(qual_ool, "q");
  While ( sqlca.sqlcode to SQL_NOF ) {
```

if ( tlistf() - SGL\_BOF ) break;

```
if ( lime >= PAGESTIE ) {
                                           /* Make sure that you have space */
           Sprintf(prm, "\f\tInventory of Columns by Table for %s\t\tPage %d\m",
                   ASSESSMENT.nune, +ipage);
          1120 = 3;}
                                                 /# Print name of the table #/
       if ( stracep (qual_ool, tid.arr, (unsigned int) tid.lem)) {
    Sprintf(prm, "\m\n44d 4s", ++j, tid.arr);
          E = 0;
          stropy(qual_ool, tid.arr);}
       else if (line = 3) fprintf(pra, "\a\a46d 4s (Gost'd)", j, tid.arr);
                                              /* Frint same of the column */
       fprintf(pra, "\a\t44d 4-15s ", ++k, cid.arr);
       lop = 0; /* Frint as many characters as the comment field contains */
       if (workspace.les - 0) (
          fprintf(pra, "\a");
          14ma++; }
       alsa {
          while (lop < workspace.lem) {
             acp = (lop+CCMACEFTSFACE < workspace.lea) ?
                        lop+Communication: workspace.len;
             for ( ; map > lap; map--) /* Find good spot to end this line */
if (workspace.arr[map] = ' ' ||
                    workspace.arr[nop] == '\0') break;
             if (lop == nop) { /* In case we have a line without white space */
fprintf(prn, " ");
                for ( ; map < lap+commerceration; map++)
                   fprintf(pra, "\t");
fprintf(pra, "\d", workspace.arr[acp]);
                   fprintf(pra, "\a\t\t\t ");
                   Idnott: 1
             else { /* Hormally we can find a white space somewhere */ workspace.arr[acp] = '\0';
                fprintf(pra, "\t%s\a\t\t\t ", &workspace.arr[lop]);
                1120++:
               lop = mop+1; }
               /# End WHILE (lop) */
         ) /* End IF(null string) */
   talista();
#1fdef CHBCBD07
alsa (
   printf("\a\a000E1 My cursor did not open!");
   AMPORAGE();
-
ist pratabo()
pratabe() --- Frint ASAN's TABLE of CONTENTS
     Routine opens cursor US then fetches rows until SQL BOF is found. 
Each row printed on the system printer. Then the cursor is closed.
      Note: Modifications to this function may impact the related
              functions vlisto(), vlistf() and vlists() that open the
              cursor, fetch rows using it respectively, and close it
************************************
Sdefine PACHETER 54
Sundaf COMMERCEDACE
#define COMMENTERACE 54
ist i, j, line, lop, sop, page;
                                            /* Counters for lines, etc.
ist vlisto();
                                             /* Database Utilities Used
int vlistf();
int vlisto():
```

if ( talistf() == SQL\_NOF ) break; /\* Get same of the next column \*/

```
SLOUT("Printing ASAM's Table of Contents");
   w = aloak();
 12 ( | vlisto() ) {
                                             /* Open systemilist query
   j = 0;
   page = 1;
    Sprintf(prm, "\tlmventory of ASAH Assessments %s\tPage %d\n", now, page);
   11ma = 2;
   while ( sqlca.sqlcode != SQL_NOT ) {
      if ( wlistf() - SQL_BOF ) break;
                                             /* Got the next assessment name */
      if ( line >= PAGESINE ) (
                                             /* Make sure that you have space */
          Sprintf(prm, "\f\tImventory of Assessments %s\tPage %d\n", now, ++page);
         1420 = 2;}
      fprintf(pra, "\a444 4-20s ", j, mid.arr); /* Assumes <= 20 char! */
      lop = 0; /* Frist as many characters as the comment field contains */
       while (lop < workspace.lem) (
         map = (lop+community)cz < workspace.lem) ?
                   lop+commerswacz : workspace.len;
         for ( ; nop > lop; nop--) /* Find good spot to end this line */
if (workspace.arr[nop] == '\0') break;
         if (lop == nop) { /* In case we have a line without white space */ fprintf(prm, " ");
                                  -};
            for ( ; acp < lcp+commercation; acp++)

fprintf(pra, "%c", workspace.arr[acp]);
               Sprintf(pra, "\a\t\t\t ");
               11no++; }
         also { /* The normal case when we can find a white space scmewhere */
            workspace.arr[ncp] = '\0';
            fprintf(prm, "%s/x/t/t ", &workspace.arr[lop]);
            line++:
            lop = nop+1; \ \}
         ) /* End WHILE (lop)*/
   ) /* End WEILE (sqlos) */

Sprintf(pra, "\n\n\t\t\t= END =\f");
   vlista();
fifdef CHECKOT
alse (
   printf("\n\noocmi My oursor did not open!");
   exposeg () ;
Seediff.
/************************************
             -- Create standard set of AEAN Assessment tables.
     This file generates:
                                      Index
                                                           Cluster
     CLUSTERS:
     TABLES:
                  R IDE
                                                           GEO CLUS
                  N_LCOM
                                                           ORDO CILTE
                  R LEQ
                                                           GEO CLUS
                  2<u>_</u>7#F
                                                           @
                  MISSIONS
                  ACTIVITIES
                  OPERATIONS
                  TANDER DO
***********************
#define SQLCA_STORAGE_CLASS extern /* Switch for header files
```

char \*now, \*clock();

```
MINIC SQL INCLUDE SQLCA;
MANC SQL REGIN DECLARS SECTION;
MING SQL INCLUDE Saharris.h;
word son menume hostware.h:
EXEC SQL END DECLARE SECTION;
                               /* Standard ASAN Seeder File
#include "asan.h"
ist istlee()
Create General Geographic cluster GMO_CLUS
***********************
   printf("\mintles ");
char *clock();
SLOUT("Reserving space for data");
HIGH SQL CHARTE CLUSTER GRO_CLUS (X_COOSD NUMBER, X_COOSD NUMBER);
#1fdef CENCROUT
 printf("\adeo_clus tld", workspace.arr, sqlca.sqlcode);
Beads C
if ( sqlom.sqloode && sqlom.sqloode |= DOFLICATE_OBJECT) {
  SLOUTES (sqlos.sqlorms.sqlorms);

fprintf(dante, "\nts ts\n\t\t\ Creating GEO_CLUS (ID = %d)",
     clock(), sqlcs.sqlerm.sqlermc, Assessment.id);
  return (int) sqlom.sqloods;}
/****************************
           Create table to receive data points of Ldm values
     Note: Only the clustered (X,Y) pairs are identified here. The
           computational routines will create the remaining (real)
           columns and decide at run time what their names will be.
**********************
EXEC SOL CREATE TABLE R_LOW (
             N COORD STREET NOT NULL, /* N-Coordinate of Point */
Y COORD STREET NOT NULL) /* Y-Coordinate of Point */
             CLUSTER CEO_CLUS (X_COORD, Y_COORD);
Bifdef CHRCHOTZ
   printf("\nR_IDM %ld", sqloa.sqloods);
Sales
  SLOUT("Log and derivatives");
fendif
if ( sqlca.sqlcode && sqlca.sqlcode |= DOFLICATE_OBJECT) (
  SLOUTER (sqlca.sqlerum.sqlerums);
  fprintf(dante, "\nes ts\n\t\t\t Creeting Table R_IDW (ID = 4d)",
     clock(), sqlca.sqlerm.sqlermc, Assisment.id);
  return (int) sqlos.sqloods;}
Create table to receive data points of Limm values
     Note: Caly the clustered (X,Y) pairs are identified here. The
           occuputational routines will create the remaining (real)
           columns and decide at run time what their names will be.
***********************
EXEC SQL CREATE TABLE R_LDOMS (
```

```
Z_COORD HUMBER HOT HULL, /* X-Coordinate of Point */
Y_COORD HUMBER HOT HULL) /* Y-Coordinate of Point */
                 CLUSTER CHO CLUS (X_COCOD, Y_COCOD);
fifdef CEBCEOUT
    printf("\RR LDWAR %ld", sqlos.sqloods);
if ( sqlos.sqloods 66 sqlos.sqloods |= DUFLICHTE_OBJECT) (
   SLOUTER (sqlos.sqlarms.sqlarmo);
    Sprintf(dente, "\m$s %s\m\t\t\t Creating Table R_LDMGR (ID = %d)",
      clock(), sqlca.sqlerm.sqlermc, ASSESSMENT.id);
   return (int) sqlom.sqloods;}
             Create table to receive data points of Leq values
      Note: Only the clustered (X,Y) pairs are identified here. The computational routines will create the remaining (real)
              columns and decide at run time what their names will be.
EXEC SQL CREATE TABLE R_LBQ (
                E_COORD NUMBER NOT NULL, /* E-Coordinate of Point */
T_COORD NUMBER NOT NULL) /* T-Coordinate of Point */
CLUSTER GRO_CLUS (E_COORD, Y_COORD);
#1fdef CRECKOUT
     printf("\nR_LBQ %ld", eqlos.sqloods);
Sendi?
if ( sqlca.sqlcode && sqlca.sqlcode != DUFLICATE_CBJECT) {
   SLOTTEP (sqlom.sqlerrm.sqlerrmo);
   fprintf(dante, "\n*s %s\n\t\t\t Creeting Teble R LMG (ID = %d)",
      clock(), sqlca.sqlerm.sqlerma, ASSESMCET.id);
   return (int) sqloa.sqloode;}
/*********************************
             Create table to receive data points of pef values
      Note: Only the clustered (X,Y) pairs are identified here. The
              computational routines will create the remaining (real)
             columns and decide at run time what their names will be.
MING SQL CREATE TABLE R PST (
                E COORD SUSSESS NOT NULL, /* E-Coordinate of Point */
Y_COORD SUSSESS NOT NULL) /* Y-Coordinate of Point */
                CLUSTER ORO_CLUS (X_COOSED, Y_COOSED);
#1fdef CERCEOUT
     printf("\nR_PSF %ld", sqloa.sqloods);
if ( sqlos.sqloods && sqlos.sqloods != DUFLICATE_OBJECT) {
   SLOUTES (sqloa.sqlarma.sqlarma);
   Sprintf (damte, "\m** *s\m\t\t\t Creeting Table R PSF (ID = *d)",
     clock(), sqlca.sqlerm.sqlermc, ASSESSMENT.id);
   return (int) sqloa.sqloode;}
            MISSIONS: Missions Flown in this Assess
***************
EXEC SQL CREATE TABLE MISSIONS (
```

```
MISSION
                        CHAR(7) NOT NOLL, /* Mission Identifier
              TYPE
                        CHAR(1),
                                       /* Mission Type
                                        /* Description of Mission */
                        CEAR (60) ,
              DESCR
                                        /* Number of A/C in sortie */
              SORTIN_SIZE MOMER);
#1fdef CENCEDUT
printf("\mmissions %ld", sqlos.sqloode);
   SLOUT("Missions and Operations");
 #end1£
if ( sqloa.sqloods && sqloa.sqloods != DUFLICATE_OBJECT) {
   SLOUTER (sqlog.sqlorum.sqlorumc) ;
   fprintf(dexte, "\n%s %s\n\t\t\t Creeting Table MISSIONS (ID = %d)",
     clock(), sqlos.sqlerms.sqlerms, ASSESMENT.id);
   return (int) sqlom.sqloode;}
ACTIVITIES: Population of Sources by Mission by Aircraft
MING SQL CHEATE TABLE ACTIVITIES (
             S_LANEL CHAR(9),
M_IDENT CHAR(7),
                                       /* Label of the "Source" */
                                       /* Identifier of mission */
              AIRCRAFT CHAR(12),
                                       /* Specific Aircraft Name */
              ACTIVITY MOMER);
                                       /* Internal Identifier
    printf("\nACTIVITIES %ld", sqlca.sqlcode);
if ( sqlca.sqlcode && sqlca.sqlcode (= DUPLICATE_CBJECT) {
   SLOUTER (sqlca.sqlerm.sqlermc);
   fprintf(dante, "\n4s 4s\n\t\t\t Creating Table ACTIVITIES (ID = 4d)",
     clock(), sqlca.sqlerm.sqlermc, ASSESSMENT.id);
   return (int) sqlos.sqloods;}
/********************************
        OPERATIONS: Number of day and might sorties by month
*************************
EXEC SQL CREATE TABLE OPERATIONS (
             ACTIVITY MOMBER,
                                /* Internal Miss/Aircraft/Route Tag */
             MOSTE
                      NUMBER (2,0), /* Month for this entry */
NUMBER, /* Daytime Operations & */
                     MORER,
             DAY
                     HOMEN,
                                 /* Blokes Flying Wight */
/* When last modified *
             LASTUYD DATE);
#1fdef CERCHOUT
    printf("\normaxioss %ld", sqloa.sqloode);
Bearing
if ( sqlca.sqlcode && sqlca.sqlcode != DUFLICATE_CBURCE) {
  SLOUTEF (sqlos.sqlerm.sqlerma);
  fprintf(dante, "\ate te\n\t\t\t Creating Table OPERATIONS (ID = td)",
    clock(), sqlca sqlerma.sqlermac, ASSESSMENT.id);
  return (int) sqlcs.sqlcode;}
/*****************************
     MIR_FLIGHT_PARAM: Flight parameters for each activity by segment
******************************
```

EXEC SQL CREATE TABLE MIR\_FLIGHT\_PARAM (

```
ACTIVITY MAGER,
                                  /* Internal Miss/Aircraft/Route Tag */
          FIX LABEL CHAR(12),
                                  /* MIR Mavigation Point Id.
                                  /* Reference for altitude
                    CEAR(3),
                                                                  */
          ALT REF
                    HOMER,
                                  /* Altitude segment is flown
          MIT
                                  /* Fower setting for the segm
          210.
                    HOOKS,
                    FREEZ,
                                  /* Speed at which it is flown
           8770
                    NUMBER (3,0)); /* Sequence number of the new point */
#1fdef CHRCHOUT
    printf("\min FLICET_PARAM %ld", sqlca.sqlcode);
1f ( sqloa.sqlooda && sqloa.sqlooda != DUFLICATE_OBJECT) {
   SLOUTER (sqlos.sqlorms.sqlorms);
   Sprintf (dente, "\nts %s\n\t\t\t Creeting Table MTR_FLIGHT_PARMM (ID = %d)",
     clock(), sqice.sqierm.sqiermc, Assisment.id);
   return (int) sqlos.sqloode;}
MER_EXP_TAB: Tabulated Leq's for each activity. For the moment
                 there is a single table. When segments can have
different flight parameters the fate of this table
                  should be reviewed.
EXEC SQL CREATE TABLE META_EXT_TAB (SIDELINE number(5,0));
stropy (workspace.arr, "CREATE TABLE MIR_EMP_TAB (SIDELINE number (5,0)) AS \
SELECT SIDELINE FROM SUPERUSER.MFR_EXP_TAB");
EXEC SQL EXECUTE DAMBIATE : workspace;
#1fdef CENCEDUT
    printf("\mMTR_EXP_TAB %ld ", sqloa.sqloods);
if ( sqloa.sqloods && sqloa.sqloods != DUFLICATE_CBJECT) (
   SLOUTES (sqlos.sqlerm.sqlermo);
  Sprintf (dente, "\ats ts\a\t\t\t Creeting Table MER_EXP_IAB (ID = 0d)",
     clock(), sqlcs.sqlerm.sqlermc, ASSESBEET.id);}
EXEC SQL DECLARE TI CURSOR FOR SELECT sideline FROM SUPERUSER.MER_EXP_TAB;
EXCEC SOL OPEN T1:
if ( sqlos.sqloods ) (
  MIDOMIG;
  SLOUTES (sqlos.sqlarm.sqlarmad);
SprintS(dante, "\n*e *s\n\t\t\t Opening MTR_EXP_ZAS (ID = *d)",
     aloak(), sqlos.sqlarm.sqlarmc, Assisment.id);}
  for (::) {
     EDCEC SQL FETCE T1 INTO :sideline;
     if (sqloa.sqloode) break;
     EXEC SQL DESERT DETO str_exp_tab (sideline) VALUES (:sideline);
     if ( sqlma.sqlmode ) {
        SLOUTEP (sqlca.sqlerm.sqlermc);
        Sprintf(dante, "\n*s *s\n\t\t\t Inserting Table MTR EXP_TAB (ID = *d)",
        clock(), sqlca.sqlerm.sqlermc, ASSESBEET.id);}
     EXEC SQL COMMET WORK;
  EDGE SQL CLOSE T1;
* QUAL_CIT1 and QUAL_CIT2 Temporary tables for citation database queries *
EXEC SQL CREATE TRRLE QUAL_CIT1 (entry_avm char(5));
```

```
fifdef CERCEDUT
     printf("\ngGAL_GIF1 %ld", sqloa.sqloode);
if ( sqloa.sqloods && sqloa.sqloods != DUFLICHTE_OBJECT) {
   SLOUTER (sqlqa.sqlarra.sqlarraq);
   Sprints (dente, "\mes &s\m\t\t\t Creeting Table QUAL_CIFI (ID = &d)",
      clock(), sqice.sqlerm.sqlermc, ASSESSMENT.id);
   return (int) sqlom.sqloods;}
EXEC SQL CREATE TABLE QUAL_CIT2 (entry_num cher(5));
#1fdef CERCEOUT
    printf("\agual_CIT2 %ld", sqlos.sqloods);
if ( sqlos.sqloods && sqlos.sqloods != DUFLICATE_CBJECT) {
   SLOUTES (sqloa.sqlorum.sqlorumc);
   return (int) sqlos.sqloods;)
return (int) sclon.scloode;
/*************************************
      missions.pc -- Routines for missions
      This file contains:
        pechanis -- Set up routine to select a new mission
        pentrais -- Set up routine to select a mission and mircraft
                       combination for a given MIR
        estermis -- Enter a mission for this assessment
        pentrilt -- Set up routine to process a mission and aircraft
                       combination for a given MTR
        permiss -- Set up routine for entering a new mission
         animispt -- Store an MCR Mavigation Point actually flows on a
        particular activity and advance all pointers.

insmispt -- Insert an MTR Havigation Point actually traversed
                       during a particular mission in the detabase table
                       of MIR segments flown
        psaddops -- Fut up the add operations window and accept data.

    Expand steady yearly operations over all months.

                  -- Varify the existence of a mission as part of an
                      assessment's definition
        canonis -- Canoel mission currently pending on the database
savemis -- Commit mission currently pending on the database
        MISCORN -- Connect an MIR with a Mission
     ORACLE access routines:
       1. ...listo() Open Cursor (Opens a "logical file")
2. ...listf() Fetch Cursor (Reads next record from logical file)
...bunch() Fetch Cursor (Reads next bunch from logical file
                       applicable only for multiple choice options)
      4. ...lists() Glose Cursor (Closes the logical file)
     WEEKE:
      ... = ss - Names of all Missions defined for this assessment
                   (Meltiple choice display option is supported)
      ... = so - Operations of a particular activity
#define SQLCA_STORAGE_CLASS extern /* Switch for header files
```

EXCEC SQL DECLUDE SQLCA;

```
RESC SQL BEGIN DECLARS SECTION;
                               /* All SQL declarations are in */
 EXEC SQL INCLUDE hostwars.h;
                               /* these header files
 EIRC SQL INCLUDE Saharris.h;
 MINE SQL MED DECLARS SECTION;
 Singlade "seen.h"
                               /* Standard ASAW Header File
 static missflg = 0;
 int pechanie ()
 pechymis -- Set up routine to select a new mission
     Noutine (1) opens oursor for fetch of all missions known (in this
            varsion missions are always local to the assessment)
(2) fetches the first betch into memory
            (3) puts up "obgounstracrees"
 ********************************
 int blakdspl(), sslisto(), ssbunch(), roode;
 #ifdef CEECHOTE
printf("\mpschgmis ");
 Send! f
 NEEREAMI (Screen, Window, Button, Datum); /* Save where you were when you */
 stropy (oldscreen, Screen);
                                  /* called this "pseudo pop-up" */
 blakdspl();
HER_SCHEEN ("chgmis");
 SLOUT("Retrieving list of missions");
roods = selisto();
return (roode ? roode : esbumch());
int postumie()
postrais -- Set up routine to select a mission and
                   aircraft combination for a given MTR
     Noutine (1) sets mission flags to false
           (2) puts up "spentrmis"
*********************************
missfly= 0;
stropy (tid.arr, ad_asso.arr);
tid.len = ac neme.len;
MEN_SCREEN("spentrais");
return (int) 0;
int entermie()
entermis -- Enter a mission for this assessment
int roods;
void emponeg();
char *clock();
Bifdef CERCEOUT
  printf("\nestermis ");
1f (ac_in_form <= 0) {
  SLOUTER ("7 Sortie sixe");
  UPDATE_DATCM("numac");}
misstype = toupper(misstype);
```

```
misdesc.lem = strlem(misdesc.arm);
EXEC SQL INSERT INTO missions (mission, type, descr., sortie_size)

VALUES (:misslabl, :misstype, :misdesc, :eq_in_form);
#1fdef CENCEOUT
printf(" %ld", sqlom.sqloods);
if (roods = (int) sqlos.sqloods) {
   expOmeg () ;
   fprintf(dante, "\m*s entermis: %s", aloak(), sqlca.sqlerm.sqlerma);
   ERRC SQL BOLLBACK WORK;
   missflg = 0;}
alse {
   MINE SOL COMMET WORK;
   missflg = 1;}
MEN SCREEN (oldscreen);
MENVALS();
retura roode:
int postrflt()
pentrfit -- Set up routine to process a mission and
                       aircraft combination for a given MCR
int cancais(); solisto(), solistf(), solistc(), vfyacutz();
wold empones();
char *clock();
                                  /* Do we know how to calculate this? */
if (vfynomir(ac name.arr))
    etura (int) sqlos.sqloods;
                                 /* Eas a mission been entered? */
if (imissflg) (
   SLOUTEF ("Please Select Mission First");
   return (int) SQL BOF; }
#ifdef CHRCHOUT
  printf("\arrevious? %s %s %s", srcid.arr, misslabl.arr, mo_mame.arr);
EXEC SQL SELECT activity FROM activities INTO :activity
               WHERE S LAREL - : srcid
AND M IDENT - :missiabl
                 AND AIRCRAFT = :ed_neme;
#1fdef CEBCHOUT
  printf(" = %ld (code = %ld)", activity, sqlce.sqlcode);
   SLOUTP ("ack");
feedif
if (sqlos.sqloods - SQL_BOF) ( /* This is a unique combination */
  EXEC SQL SELECT MAX(activity) FROM activities INTO :activity;
   #1fdef CEECROUT
     printf("\nLast activity = %ld code = %ld", activity, sqlca.sqlcode);
   feedic
   if (sqlos.sqloods)
     if (sqloa.sqloods -- SQL_BOF) activity = 0;
     also (
        fprintf(dente,"\ats Pfmtrflt2: ts", aloak(), sqloa.sqlerms.sqlerms);
        return (int) sqloa.sqloode;}
  activity++;
  EXEC SQL INSERT DFTO activities (S_LABEL, M_DDEST, AIRCRAFT, ACTIVITY) VALUES(:srcid, :misslabl, :so_meme, :activity);
  #1fdef CENCROUT
     printf(" Insert = %ld", sqlca.sqlcode);
     SLOUTP ("ack");
   Annet14 5
  if (sqlos.sqloods) {
     exponent();
     fprintf(dante, "\n4s Pfstrflt3: 4s", clock(), sqlca.sqlerma.sqlermac);
     return (int) sqlca.sqlcode; }
  curnavpt.arr[curnavpt.len = 0] = '\0';
  prenavpt.arr[prenavpt.lem = 0] = '\0';
```

```
lptr1 = &curlowait.altitude;
    optri = curlowalt.units;
    MENVALS ();
    MRW_SCREEN ("mtrflt");
 elee (
    if (sqlom.sqloods) {
       emp(mang ();
       fprintf(dente, "\nts Pintrflt1: ts", clock(), sqlca.sqlerm.sqlermc);
         ; () eteoe
       return (int) sqlom.sqloods;}
    solisto();
    if (sqloa.sqloods) {
        amponeg () ;
        sprintf(workspace.arr, "FAILED - %s", sqlca.sqlarm.sqlarmc);
        Sprints(dants, "\n\s OFEN \s", alock(), workspace.arr);
Sprints(dants, "\n\t\t\t Retrieving Operations for activity \ld",
                       ectivity);
        concents ();
        return (int) sqlos.sqloods;)
    for (;;) {
       solists();
       if (sqlos.sqloods) {
          if (sqlos.sqloods != SQL_BOF) {
             amponeg():
             sprintf(workspace.arr, "FAILED - %s", sqlos.sqlorm.sqlorms);
fprintf(dante, "\n%s SELECT %s", clock(), workspace.arr);
fprintf(dante, "\n\t\t\t Retrieving Month %d Operations for %ld",
                             active_mo, activity);
             communic ();
             1
       break;
       ops[active_mo-1].day = ops_day;
ops[active_mo-1].mite = ops_mite;
   if (sqlos.sqloods - SQL_BOF) (
       solista();
       EXEC SQL DELETE FROM operations WHERE activity = :activity;
       fifdef CEECHOUT
         printf("\nPrevious data deleted %ld", sqlom.sqloode);
          SLOUTP ("ack");
       Sendid
   else solists();
   ADD_WIMDOW("month", 7, 1);
   SLOUTP ("You already have data!");
return (int) 0;
int penumien (neme)
penumish -- Set up routine for extering a new mission
      Noutine (1) verifies that the new name is unique.
               (2) puts up the mission definition screen and starts it
************************
char zemo[];
char *clock();
int roods, melists(), mebunch();
void exponeg();
Bifdef CERCHOUT
printf("\npenumien ");
fendif
REMOVE_WINDOW();
SLOUT("Processing New Mission Request");
if (( reads = vfyEmis(name)) != SQL_EOF) { /* Check for strange things */
   1f ( roods - 0) (
      sprintf(workspace.arr, "Sorry, but is already exists", name);
SLOUTEP(workspace.arr);
```

```
return (-1);}
  empomeg();
   fprintf(dante, "\nts Pfnewmis: ts", clock(), sqlom.sqlerms.sqlerms);
  return roods;}
alsa {
                                   /* Close list of choices
  seliste():
                                  /* This is the new mission */
  stropy (misslabl.arr, m2bv.arr);
  misslabl.lea = m2bv.lea;
  misdesc.lem = 0;
  misdesc.arr[0] = '\0';
  MEN SCREEK("misspec");
  HENVALUE ():
   MENT_DATOM("misdesd"); Removed for possible conflict with U Bug */
  return (int) 0; }
ist astmispt()
arthispt -- Store as MTR Newigetion Point actually flows on a
               particular activity and advance all pointers.
***************************
register int 1;
int insmtrpt();
Sifes CERCHOUT
printf("\anstaispt ");
.
Jendif
if (linesispt())(
    /* Advance the May Point Parameters */
  for (i = 0; i < 2; i++) prenavpt.arr[i] = curnavpt.arr[i];
for (i = 0; i < 10; i++) prelowalt.spec[i] = curlowalt.spec[i];</pre>
  so_pre_spd = sc_cur_spd;
  ad pre per - ad der per;
   curnavpt.arr[0] = '\0';
  MENVALS ();
  MEXT_DATOM("ournevpt1");
return (int) sqloa.sqloods;
}
int immisot()
/*****************************
     insmispt -- Insert as MTR Havigation Point actually traversed
                 during a particular mission in the database
****************************
register int 1;
roid exposes ();
#1fdef CERCEOUT
printf("\minemispt ");
stropy(a2bv.arr, sroid.arr);
stroat (n2bv.arr, curnavpt.arr);
n2bv.len = strlen(n2bv.arr);
fitseg += 1:
EXEC SQL IMEERS INTO MER_FLIGHT_PARAM
                     (ACTIVITY, FIX_LABEL,
                                             ALT_REF.
                                                        M/T.
                      2700.,
                                 SPD,
                                              SEQ)
             VALUES ( :activity, :a2bv,
                                             :aptr1,
                                                         :lptr1,
                      :so_our_pwr, :so_our_spd, :fltseq);
#1fdef CEECEOUT
 printf(" Insert = %ld", sqloa.sqloode);
```

```
if (sqlos.sqloods) {
    if (sqlca.sqlcode -- EXISTS) SLOUTES ("Duplicate Entry");
   else amponsg();
canomis();} /* This is necessary since ORAGLE automatically rolls back */
/* the inserts already done at this point ! */
 return (int) sqlos.sqloods;
 int pseddops (count)
 psaddops -- Put up the add operations window and accept data.
 ******************************
 if (count = 1) ADD_WINDOW("daymite", 6, 1);
else if (count = 12) ADD_WINDOW("month", 6, 1);
else ELOUTP("PEADDOPS called with bed argument");
 return (int) 0;
int expmops()
 -- Expend steady yearly operations over all months.
register int 1;
for (1 = 1; 1 < 12; 1++) {
   ops[1].day = ops[0].day;
ops[1].mite = ops[0].mite;)
return (int) 0;
int viyinis (name)
vfyRmis -- Verify the existence of a mission as part
     Nowtine looks in the assessment's MISSIGHS table, loads mission
     into : aid and returns sqloa.sqloods for the query.
******************************
cher zeme[];
register int 1;
#1fdef CHBCHOUZ
printf("\avfyMais ");
Bendie
m2bv.len = strlen(name);
for ( i=0; i < n2bv.len; i++)
   n2bv.arr[i] = toupper(name[i]);
a2bv.arr[1] = '\0';
THE SQL SELECT mission
       FROM missions
       INTO
            : 014
       WEERE mission = :n2bv;
did.arr[did.len] = '\0';
#1fdef CEBCEOUT
  printf(" %ld ", sqlom.sqloode);
```

```
retura (int) sqlom.sqloode;
int concess ()
      canonis -- Cancel the mission currently pending on the datab
    Routine always issues a ROLLBACK
int postrent();
#15def CERCHOUT
printf("\mommas ");
Seeds C
NORC SOL BOLLBACK WORK:
#1fdef CERCHOOT
  primtf("%ld", sqloa.sqloods);
Seedif.
sprintf(workspace.arr, "Entry for mission 4s CANCHLED", misslabl.arr);
SLOUISE (workspace.arr);
missiabl.orr[missiabl.lem = 0] = '\0';
pentrent();
return (int) sqlom.sqloods;
int sevenis()
sevenis -- Commit the mission currently pending on the detabase
int mtrtabl(), pentrent();
Fifder CHRCHOUT
printf("\msavemis ");
for (active_mo = 1; active_mo < 13; active_mo++) {
   ope_day = ope[active_mo-1].day;
   ope_mite = ope[active_mo-1].mite;
   EURC SQL DESERT DETO operations ( ACTIVITY, MOSTE, DAY, MIGHT, LASTOFD)
          VALUES (:activity, :active_mo, :ops_day, :ops_mite, SYMDATE);
   if (sqlca.sqlcode) {
     EXEC SQL NOLLBACK WORK;
     postrost();
     return (int) sqlom.sqloode;)
EXEC SQL COMMET WORK;
mtr_tabl();
pentrest();
return (int) sqloa.sqloode;
int MISconn (name)
/**********************************
```

```
MISconn -- Connect as MIR with a Mission
       Nowtine SELECTS the Mission from the detabase and loads parameter
       block. Returns an error code which indicates whether or not
       the SELECT was successful.
       Note: The existence of the mission on the database is assumed.
              SQLCA.SQLCODE is returned and if monsero am error
              message is displayed on the status line.
 char zeme[];
 ł
 register int 1;
 int NEW_SCREEK(), MENVALS(), SLOUT();
 char *clock();
 Bifdef CERCHOTE
  printf("\mMISconn ");
 Seedil!
SLOUT("Retrieving Mission");
n2bv.len = strlen(name);
 for (i = n2bv.len; i >=0; i--) n2bv.arr[i] = name[i] = toupper(name[i]);
 EXEC SQL SELECT mission, type, descr, sortin_size
         FROM missions
1970 :misslabl, :misstype, :misdeso, :ac_in_form
         WHERE mission = :n2bv;
 #1fdef CHBCROOT
  printf("%ld ", sqlom.sqloods);
 Beadle
 if (!sqlca.sqlcode) {
                                        /* Everything is O.E. */
   misslabl.urr[misslabl.lem] = '\0';
   misdesc.arr[misdesc.len] = '\0';
   seliste();
   missellg = 1;
   HEWVALS();
   MER_SCREET (oldscreen) ;
              /* This should never happen! But, .... */
   missflg = 0;
   EDOMA:
   sprintf(workspace.arr, "FATLED - %s", sqlom.sqlorrm.sqlorrmo);
fprintf(dante, "\n%s SELECT %s", clock(), workspace.arr);
fprintf(dante, "\n\t\t\ Retrieving Mission %s", n2bv.arr);)
return (int) sqlos.sqloods;
           selisto --- Open cursor $1 for a list of missions that
                     belong to the current assessment
     Routine executes an open oursor command for oursor $1 and them
     returns to the calling program with the CRACKE status code.
     Note: Modifications to this function may impact the related
             functions selistf(), ssbunch() and seliste() that fetch
             rows and close the cursor and, possibly, functions that
             cell these utility routines.
#1fdef CHICKOUT
printf("\meslisto ");
EXEC SQL DECLARE S1 CURSOR FOR SELECT mission
                         FROM missions ORDER BY mission;
EXEC SQL OPEN S1;
#ifdef CRECEOUT
printf(" Open: %ld ", sqlcm.sqlcode);
```

```
return (int) sqloa.sqloode;
int selistf()
/-----------
     selistf --- Fetch a row using the opened cursor $1 for missions
     Nortine executes an fetch command for cursor $1, which is assumed
     to have been opened, and them returns to the calling program with
     the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions selisto(), sebunch() and selista() that open,
            fetch groupwise and close the cursor and, likely,
            functions that call these utility progress
EXEC SQL FETCH S1 13FFO :misslabl;
misslabl.arr[misslabl.lem] = '\0';
Bifdef CHRCHOTT
printf(" Fetch: $1d ", sqlos.sqloods);
return (int) sqlos.sqloods;
int sebunch()
           --- Fetch a bunch (20 or whatever the size of deplocalt)
                  using the opened cursor $1 for mission list
     Routine executes an fetch command for cursor $1, which is assumed
     to have been opened, and them returns to the calling program with
     the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions selisto(), selistf() and selista() that open,
            fetch and close the cursor and, likely, functions that
            call these utility programs
register ist 1;
int prove, prove, roode;
char *clock();
EXEC SQL FETCE S1 INTO :deplault;
arows = (int) sqloa.sqlerrd[2];
mrows = (sizeof deplault) / 34;
roode = (int) sqlqq.sqlqode;
#ifdef CENCEDUT
printf(" seBunch: %ld returns %d of %d rows", sqlca.sqlcode, arows, arows);
Seedil 2
if ((roods = SQL FETCE OUT OF ORDER) || (roods = SQL BOF)) {
  if ((roods = SQL BOF) && (nrows > 0)) {
     for (i = 0; i < arous; i++) deplant[i].arr[deplant[i].lea] = '\0';
     if (arows < arows)
      for (i = arows; i < arows; i++) deplacit[i].arr[0] = '\0';
    MENVALS ();
     SLOUTS ("The last mission in the list is on the screen"); }
  else {
    HENVALS ();
     SLOUTP ("You are already as far down in the list as you can go"); }
alse {
  if (roode) {
    ENDOMES;
    fprintf(dante, "\nes sssusce: es", clock(), sqlca.sqlerm.sqlerme();}
  return roods;
```

```
ist seliste()
 selists --- Close cursor $1 for mission list
      Routine executes a close cursor command for cursor $1 and them
      returns to the calling program with the CRACLE status code.
      Note: Modifications to this function may impact the related
             functions selisto(), sebunch, and selistf() that open the
             cursor and fetch rows using it and, possibly, functions
             that call these utilities
 ************************
 #1fdef CHRCHOUT
printf("\meslista ");
 .
Seedif
 HARRO SQL CLOSE S1;
 Bifdef CHRCHOUT
printf("Close: tld ", sqloa.sqloode);
 dendi f
 return (int) sqloa.sqloode;
int solisto()
solisto --- Open oursor $2 for a list of operations that
                    belong to the current activity
     Rowtine executes an open cursor command for cursor $2 and then
     returns to the calling program with the CRACKE status code.
     Note: Modifications to this function may impact the related functions solists() and solists() that fetch rows and
             close the cursor and, possibly, functions that call
             these utility routines.
#1fdef CENCEDUT
printf("\msolisto ");
-
EXEC SQL DECLARE S2 CURSOR FOR SELECT month, day, might, lastupd
            FROM operations REERE activity = :activity ORDER BY month;
#1fdef CERCEOUT
printf(" Open: %ld ", sqloa.sqloods);
fendis
return (int) sqlom.sqloods;
           ,
《我也我的我也我也是我们我们的我们的我们的,我们就是我们的我们的我们的我们的我们的,我们就是我们的我们的我们的我们的我们的,我们就是我们的,我们就是我们的,我们
     solistf --- Fetch a row using the opened cursor #2 for operations
     Rowtine executes an fetch command for oursor $2, which is assumed
to have been opened, and then returns to the calling program with
the ORACLE status code.
    Note: Modifications to this function may impact the related
            functions solisto() and solisto() that open and close
            the cursor and, likely, functions that call these utilities
EXEC SQL FETCE 82 INTO :sotive_mo, :ops_day, :ops_mite, :timest2;
```

```
timest2.arr[timest2.lem] = '\0';
 #1fdef CHRCROTT
 printf(" SOlistf: %ld ", sqlom.sqloods);
 Seedif.
 return (int) sqlos.sqloods;
 int solists()
 /***********************************
          solists --- Close oursor $2 for operations list
     Routine executes a close cursor command for cursor $2 and then
     returns to the calling program with the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions solisto() and solistf() that open and fetch
            rows from the cursor and, possibly, functions that call
            these wtilities
 printf("\msoliste ");
 -
BOOK BOL CLOSE 82:
Fifdef CERCEOUT
printf("Close: %ld ", sqloa.sqloods);
 retura (int) sqlca.sqlcode;
mtr.pd -- Routines calculating noise exposure from MTRs.
     This file contains:
     isperpen - Return distance of point to line if perpendicular is
                inside line segment, NULL otherwise
     g streel - Got the 2-parameter SEL distance approximation for
                an aircraft
     mtr_seel - Calculate the Single Event Exposure Level for an MTR
     mir tabl - Calculate the noise/distance table for an MTR segment
                when the computational parameters have been loaded
            - Connect as assessment to as MIR
/* Seader for calls to MM-DOS
#include coroses.h>
#incinde <stdio.b>
finclude casth.h>
#define SQLCA STORAGE CLASS extern
                               /* Switch for header files
EXEC SQL REGIN DECLARS SECTION;
                               /* All SOL declarations are in
EXEC SQL INCLUDE hostwars.h;
                               /* these header files
EXEC SQL DECLEDE Saharris.h;
VARCEAR rowid[25];
EXEC SOL END DECLARE SECTION:
MINEC SQL INCLUDE SQLCA:
                               /* Standard ASAN Header File
double delta_pwz;
                   /* Global: adjustment for actual power setting
double delta_spd;
                    /* Global: adjustment for actual aircraft speed */
double 20, PYO:
                   /* Global: coordinates of point of observation
                    /* Global: goordinates of a segment end point */
double X1, FY1;
double 12, 12;
                   /# Global: coordinates of a segment and point #/
double isperpes()
isperpen -- Routine to determine if the perpendicular of
```

```
Routine performs a translation of the coordinate system so that
              (EO, PTO) is at the origin of the new coordinate system. It them
              rotates this new goordinate system around (EO.FTO) to a second
              new coordinate system in which the X-axis is parallel to the line
             segment. If the perpendicular to the line segment is inside the segment, then in this occurringte system the X-occurringtes of the
              and points of the segment will have different signs. In this
              system, the absolute value of the Y-coordinate of either point is
              the distance from the point to the segment.
            Notes: 1. If the Y'' value (in the second coordinate system)
                                        is positive, a person moving from (21,271) to (22,72) will observe (20,270) as being to his/her right.
                                2. The original coordinates are (XI,FYI), etc.
                                         The translated coordinates are (Xip, Yip), etc.
                                         The translated and rotated coordinates are (Midp, Midp)
                                         etc. to mimic the usual notational convention of
                                        primed and double-primed coordinate pairs.
double Mp0, Mip, M2p, Mp0, Mip, M2p;
double Mdp0, Midp, M2dp, Mdp0, Midp, M2dp;
 double sqrt();
 double hypothenuse, sinetheta, cosinetheta;
#1fdef CEBCECOT
printf("\misperpen ");
 -
hypothenuse = sqrt( (X2 - X1)*(X2 - X1) + (X2 - XY1)*(X2 - XY1) );
sinetheta = (X2 - X1) / hypothemuse;
cominetheta = (X2 - XX1) / hypothemuse;
Mip = M1 - M0;
Tip = FY1 - FY0;
122p = 122 - 20;
12p = 12 - 210;
Hidp = Hip * cocinetheta + Yip * sinetheta;
Yidp = -Hip * sinetheta + Yip * cocinetheta;
Hidp = Hip * cocinetheta + Yip * sinetheta;
Y2dp = -X2p * sinethets + Y2p * cosinethets;
#1fdef CHECKNOT
printf("\nSin = $7.5f Cos = $7.5f Hyp = $6.2f",
              sinetheta, cosinetheta, hypothenuse);
printf("\mm0 = (45.1f,45.1f)", m0, pr0);
printf("\nx' = (45.1f, 45.1f) Y' = (45.1f, 45.1f)", XI, FTI, X2, Y2);

printf("\nx' = (45.1f, 45.1f) Y' = (45.1f, 45.1f)", XIp, XIp, XIp, XIp);

printf("\nx'' = (45.1f, 45.1f) Y'' = (45.1f, 45.1f)", XIdp, XIdp,
feedif
return ( ( (XIdp > 0) && (X2dp > 0) ) ||
( (XIdp < 0) && (X2dp < 0) )
                  7 MOLL
                  : fabs (Y2dp) );
}
                g_mtreel -- Routine to get the MTR two-parameter approximation
                                                to the SEL - Distance curve from the detabase
            Routine selects the proper profile from the database and loads
            the host variables. It returns the sqlos.sqloods of the select.
               Note: An error will result if the entries in the database are
                              not unique! At this time no provision is made for a
                              pick-and-choose approach to SEL profiles.
```

```
int temp;
#15def CERCEOUT
printf("\mg_mtr_sel ");
 .
Pendif
EXEC SQL SELECT power_units, power,
               slope, intercept, speed
:pr_pur_u, :pr_power, :pur_scale,
                :pwr_slope, :pwr_intopt, :pr_speed
               strealtab
        TROM
        WHIRE aircraft = :ac name;
#1fdef CEECHOTE
fprintf(prn, "\nts tlf to speed = tlf, sqloods tld\f",
     ad_name.arr, pr_power, pr_pwr_w.arr, pr_speed, sqlos.sqloods);
printf("tld", sqlom.sqloode);
feed1£
retura (int) sqlos.sqloods;
double mtr_seel(lateral)
mtr_seel -- Routine to calulate moise exposure at a point that
                    is lateral to an MFR for a SIRGLE specific activity
                    on that MTR. (Operational adjustments not made.)
double leteral:
                               /* Leteral distance to track */
double log_magle;
                               /* LOG10 (angle above the horizon) */
double osset:
                               /* Rise time parameter
                                                               */
double onset pen;
                               /* Penalty assoc'd therewith
                               /* SEL adjusted for power and speed */
double stan2(), log10(), pow(); /* math.h routines used
          /* Calculate "Fure" Propagation-adjusted SEL */
#1fdef CHECKOUT
  printf("\nAltitude %1f Leteral %1f", ac_alt, lateral);
  log_angle = 1.758122632 + log10(stan2(sc_alt, interal));
seledi
        - pwr istopt
           + pwr_slope * ( 0.5 * log10 (ac_alt*ac_alt + lateral*lateral) )
            + delte_pwr
                                      /* Fower Adjustment
           + delta_spd
                                       /* Speed Adjustment
           - (log_angle < 1.477121255 7 /* Ground adjustment if < 30 deg */
                 (6.995 - 6.606 * log_angla
+ 1.566 * log_angla * log_angla) : 0.0);
           /* Calculate Rise Time Femalty */
oaset = 100.0 / (1+ pow(2.7183, 10.01 - 3.62 * log10(ac_speed)
                             + 2.48 * log10(ac_alt)
                             + 0.15 * log10(lateral)
                              ~ 0.0542 * seledj));
                      omset_pem = 0.0;
omset_pem = 16.6 * log10(omset/15.0);
if (caset < 15.0)
else if (omset <= 30.0)
                       oaset_pea = 5.0;
وعتو
 fprintf(prm, "seledy $10.41f caset $10.41f", seledy, caset);
Sendif.
return (saladj + caset_pen);
```

int mtr\_tabl()

```
mtr_tabl -- Routine to calulate noise exposure - perpendicular
                    distance table (MER_EXP_TAB) for a specific activity *
 ***********************
***
      NOTE: This routine currently assumes that operations parameters ***
***
              are fixed over an MTR. It must be changed when that is - ***
             no longer true. Also INTLEE.PC must them be changed!
                                                                   ***
 *************************************
char *clock();
int alisto(), alists(), alista(), i;
                     /* Finds profile for this aircraft
      g mtrsel():
double mtr_seel();
                          /* Single event calculation routine
 void exponeg();
SLOUT("Calculating METR Exposure Table");
stropy(tid.arr, "MFR_EXP_EAB");
tid.lea = 11;
sprintf(a2bv.arr, "ACT+04ld", activity);
m2bv.lem = strlem(m2bv.arr); /# Just im case activity > 9999 */
This query only computes things for the 1st segment. If we start
    allowing different operations parameters per segment, then this
    must be changed. Also see the note in INTLEM.PC!
ELECT alt, pwr, spd FROM

DETO :ac_alt, :ac_powar, :ac_speed

HERES activity = :activity AND seq = 1;
EXEC SQL SELECT alt,
                                         FROM mtr_flight_peres
fifdef CERCEOUS
  printf("\maircraft is $1f, $1f Ers, power = $1f sqloods = $1d",
   ac_alt, ac_speed, ac_power, sqloa.sqloode);
fprintf(prm, "\AAircraft is %1f, %1f KPS, power = %1f sqloods = %1d",
           sq_slt, sq_speed, sq_power, sqlca.sqlcode);
Fondif
clisto(); /* Open detabase cursor */
for (;;) {
   olists();
             /* Fetch the next instance */
  if (isqlos.sqloods) {
     cid.arr[cid.len] = '\0';
     if (!stromp (cid.arr, n2bv.arr)) break;
also cid.arr[0] = '\0'; cid.len = 0;}
                                            /* Have it already! */
                                            /* No, not that one */
   else if (sqlos.sqloods we sqt_MOF) breek;
                                            /* Have to dreste it */
  else {
                                            /* Trouble!
     fprintf(dante, "\nts %s", clock(), sqlca.sqlarm.sqlarmo);
fprintf(dante, "\n\t\t\t Retrieving MTR_EXP_TAB %s", n2bv.arr);
     SLOUTEF ("Calculation Aborted");
     GloseORA();
     return (int) sqloa.sqloode; }
alista();
             /* Close database carsor */
if (!cid.len) {
                        /* Create a new column in the table */
  sprintf(workspace.arr, "ALTER TABLE MIR END TAB ADD (to MOMER)", a2bv.arr);
   workspace.len = strlen(workspace.arr);
  fifdef CEECHOUT
    printf("\nts", workspace.arr);
fprintf(prn, "\nts", workspace.arr);
  Sendif.
  EXEC SQL EXECUTE DESIDIATE : workspace;
     printf("\n&lter Table %s &ld", n2bv.arr, sqlca.sqlcode);
     fprintf(prn, "\nklter Table to Sld", n2bv.arr, sqlca.sqlcode);
  Sendif.
  if (sqlos.sqloods) (
     exponey();
    fprintf(dante, "\n*e %s", clock(), sqlca.sqlerms.sqlerms);
fprintf(dante, "\n\t\t\t Creating MTR_EMP_TAB %s", n2bv.arr);
    SLOUTES ("Calculation Aborted");
```

```
closeORA():
       return (int) selon.seloods;
    sprintf (workspace.azz,
     COMMENT OF COLUMN META EXP_EAR. As IS 'Operations on he by he on mission he's,
    a2bv.arr, sroid.arr, so_neme.arr, misslabl.arr);
    workspace.len = strlen(workspace.arr);
    BASSAS CHRCHOUT
      printf("\n0s", workspace.err);
    EXEC SQL EXECUTE MARROTATE :workspace;
g_mtreel();
 delts_pwr = (sd_power - pr_power) * pwr_scale;
delts_spd = 10.0 * log10(pr_speed / sd_speed);
 sprintf (workspace.arr,
     "SELECT rowld, sideline, he FROM mir_amp_tab FOR UPDATE OF he",
     a2bv.arr, a2bv.arr);
 workspace.len = strlen(workspace.arr);
#1fdef CENCEOUT
   printf("\aPREPARE %s", workspace.arr);
Seedif.
EXEC SOL PREPARE D1 FROM : workspace;
Bifdef CHBCROUT
   printf("\n returns %ld", sqlca.sqlcode);
    fprintf(pra, "\ats returns tld", workspace.arr, sqlos.sqloods);
 Bendif
EXEC SQL DECLARE C1 CURSOR FOR D1;
 EXEC SQL OFER C1;
 Sifdef CHECKOUT
   printf("\nopen returns %ld", sqloa.sqloods);
    fprintf(prm, "\nopen returns %ld", sqlom.sqloods);
Seed12
for (;;) {
    EXEC SQL FETCE C1 ISTO :rowld, :sideline, :exposure;
      printf("\nFetch sideline tlf returns tld", sideline, sqlca.sqlcode);
       Sprintf (pra,
       "\nFetch sideline tlf returns tld", sideline, sclos.scloode):
   if ((sqlos.sqloods) && (sqlos.sqloods != NULL_FETCHED)) break;
         rure = mtr_seel(sideline);
   if (rowid.les != 18) {
      fprintf(pra, "\nrowid length = %d", rowid.len);
rowid.len = 18;
   sprintf (workspace.arr,
          "UPDATE MIR EXP_TAB SET to = tif WHERE rowld = 'te'",
         m2bv.arr, exposure, rowid.arr);
   workspace.len = strlen(workspace.arr);
   EXEC SQL EXECUTE DESCRIPTE : workspace;
   #1fdef CEECHOUT
      printf("\nts REPURMS tld", workspace.arr, sqlca.sqlcode);
      fprintf(pra, "\nEXE Dat to RETURNS tid", workspace.arr, sqloa.sqloade);
if (sqloa.sqloods == SQL_BOF) ( /* We finsished the list */
   EXEC SQL COMMIT WORK;
   emponeg();
   fprintf(danta, "\nes %s", clock(), sqlca.sqlerrm.sqlerrmc);
fprintf(danta, "\n\t\t\Tetching MTR_EUF_TAB for schivity %ld %lf",
             activity, sideline);
   EXEC SQL ROLLBACK WORK;
EXEC SOL CLOSE C1:
int MTRooms (name)
          MIRCORN -- Connect an ASAN assessment with an MIR
```

150

```
Noutine SELECTS the MTR from the detabase and loads parameter
       block. Returns an error code which indicates whether or not
        the SELECT was successful.
       Note: The existence of the MTR on the detabase is assumed. If
               any SQL error is found, the routine returns sqlom.sqloods
               with an error on the status line. When the MTR exists
               the MTR Data Entry Screen will be displayed. To varify
              the existence of an MCR use viyintr().
 thar asse[]:
 register int i:
 ist MEN_SCHEEN(), MENVALS(), SLOUZ(), SLOUZER();
 char *clock();
 #1fdef CHRCHOUT
   printf("\nMTRooms ");
 Bear 44.0
 SLOUP("Switching NETRS ....");
 m2bv.len = strlen(neme);
 for (i = a2bv.len; i >=0; i--) a2bv.arr[i] = name[i] = toupper(name[i]);
 EMBC SQL SELECT label, status, type, descr, orig, sched, owner, TO CHAR(date pub, 'dd-Mon-yyyy'),
                TO CEAR (timestump, 'dd-Mon-yy EE24:MT:88')
         PROM
         NETO :sreid, :srestat, :sretype, :sredesc, :sreorig, :sresched,
                :sroowser, :sropdate, :sroedate
         WHERE label = :a2bv AND type = '1';
 #1fdef CENCEDOT
   printf("%ld ", sqloa.sqloode);
Beadle
 if (!sqlca.sqlcode) {
                                         /* Everything is O.E. */
   sroid.arr[sroid.len] = '\0';
sroidsc.arr[sroidsc.len] = '\0';
sroorig.arr[sroorig.len] = '\0';
   srosched.arr(srosched.les] = '\0';
   stropy(laststr.arr, sroid.arr);
   lastatr.lea = srcid.lea:
   melieta();
   minstruct ();
   NEW_SCREEN ("mtratry");
   MENVALS ();
   if ((erestat != 'A')) {
      sprintf(workspace.arr, "Note: Status of this MER is \"to\"", srostat);
      SLOUIS (workspace.arr);}
else {
              /* This should never happen! But, .... */
   EMPONES -
   sprintf(workspace.arr, "FAILED - %s", sqlca.sqlarma.sqlarmac);
   if (sqlca.sqlcode != sqt_BOF) stours (workspace.arr);
   also SLOUTER ("This MTR exists but is not available to you");
   fprintf(dante, "\n*s SELECT %s", dlock(), workspace.arr);
fprintf(dante, "\n\t\t Retrieving MFR %s", n2bv.arr);
return (int) sqloa.sqloode;
screensa.pc -- Routines to set up the screens for assessments.
     This file contains:
     loggedon - Answers the question: "Did the planner log on?"
     pschhaak - Puts up database housekesping screen
     pechgass
               - Futs up change assessment some
               - Puts up new assessment screen
       eprobet
                - Puts up problem status sureem
     binkdspi - Blanks out the common display area
******************************
finclude cd.molude
                                    /* Seader for calls to MS-DOS
#include <stdio.b>
#define SQLCA STORAGE CLASS extern /* Switch for header files
REEC SQL INCLUDE SQLCA;
```

```
/* All SQL declarations are in
 EDIEC SOL REGIN DECLARE SECTION:
                                                /* these header files
 EXEC SQL INCLUDE hostvars.h;
 EXEC SQL INCLUDE Saharris.h;
 MING SQL MED DECLARS SECTION;
                                               /* Standard ASAN Seeder File */
 finclude "agen.h"
-- Routine to determine if the pleaser filled in the
                             username on the first screen.
 #define CEAR SERO '0'
 cher *clock();
 int i, j, closeCRA();
 #1fdef CENCHOUT
 printf("\mloggedom ");
 feadif
 MEEREMAI (Screen, Window, Datum, Button);
 fifdef CERCHOUT
printf("called from %s", Screen);
 plantnem.lea = strlea(plantnem arr);
                                                     /* Who is this person? */
 for (i = 0; i < planram.len; i++) (
if (planram.arr[i] > CEAR_EERO) break;
     plearnen.lea--;
     for (j = 0; j < plantam.len; j++) plantam.arr[j] = plantam.arr[j+1];
#1fdef CEBCROOT
printf(" Username = %s is %d characters", planznam.arr, planznam.lem);
  كناءها
 if ( plearnem.les == 0 ) { /* Did not enter a name! (or we lost it) */
    if (stromp(Screen, "firstscreen")) {
       (stromp(Strome, "firststromen")) {
SLOUTY("TROUBLE! ASAM has forgotten your name....");
Sprintf(dante, "\n\*s Logsedon: ASAM \"lost\" planner's name", clock());
Sprintf(dante, "\n\t\t\t ASSESMENT.name was \*s", ASSESMENT.name);
Sprintf(dante, "\n\t\t\t Strome \*s", Stromen);
Sprintf(dante, "\n\t\t\t Strome \*s", Stromen);
Sprintf(dante, "\n\t\t\t Datum \\*s", Datum);
Sprintf(dante, "\n\t\t\t Surton \\*s", Datum);
Sprintf(dante, "\n\t\t\t Sutton \\*s", Datum);
       MEN_SCREEN("firstscreen");
       closeCRA();
       amit (255) ;
       return (int) 255;
       )
       SLOUTS ("I don't know yet who you are. Please enter your name");
       UPDATE_DATUM("planramm");
       VCAPITAL (Splanman) ;
       ADD_WINDOW("password", 15, 3);
       UPDATE_DATUM("password");
       VCAPITAL (&password) ;
       REMOVE_WINDOW();
       }
   return (int) SQL_BOF;
return (int) 0;
int pschieck()
             pedbheak -- Set up routine for housekeeping screen
      Routine (1) makes sure that user is known
```

The second secon

```
(2) makes an entry in the audit file
            (3) puts up "dobsekpgsozeen"
int closeCRA(), loggedon();
cher *clock();
loggedon();
Sprints (deste,
"\a\n\n*s PEDRICK: %s signed on as SUFEWEER for File Maintenance\n\n*, clock(), plannum.err);
if (stromp(ASSESSMENT.name, "SUFEWEER") ) {
  1f (STooms()) {
     Sprintf(dante, "4s PEDRESEK: SUPERUSER connect failed\a\a", clock());
     Sprints(dante, "\m\t\t\ #s", sqlos.sqlorm.sqlormso);
SLOUTED("SUPERUSER access demied. Not good.....");
     aloseORA();
     exit (256) ;
     }
HHM_SCREEN("dbhsekpgscreen");
return 0;
pachgass ()
/***************************
         pechgase -- Set up routine for change assessment scree
     Boutine (1) opens cursor for fetch of CRACLE usernames
            (2) fetches the first batch into memory
            (3) puts up "abgaurassareen"
int blakdspl(), wlisto(), wbunch(), roods;
#1fdef CEECEOUT
printf("\mpschgass ");
.
Seedif
blakdepl();
MEN_SCREM("chgouresscrees");
SLOUT("Retrieving ASAN's table of contents");
roode = ulisto();
if (Iroods) roods = ubunch();
return roode;
peawasen -- Set up routine for new assessment screen
     Routine (1) verifies that the new name is unique.
            (2) creates the new ORACLE user and tables for this
               assessment.
    Notes: 1. When routine determines that it is stuck, closeCRA
               is called to terminate the execution.
            2. Execution is terminated when this routine is called
               without being connected to ORACLE. (One should not
               be able to get ossealf into this predicament.)
cher zemo[];
int roods, ASAMconn(), intlse(), arolloga(), SUccen();
int ulisto(), ubunch(), ulisto();
wold exposes ();
#1fdef CEECEDUT
```

```
printf("\npsnwasea ");
  Seedif.
    OVE_WINDOW();
 SLOUZ ("Processing New Assessment Request");
 if (( roods = vfyOid(name)) != SQL_BOF) { /* Check for strange things */
    if ( roods - NOT_LOGOND_ON ) | roods - SQL_NAD_LOGON) {
       sprintf (workspace.arr,
           "PSnewass: You are not connected to CRACLE! -- FATAL");
       SLOUTS (workspace.arr);
       fprintf(dente, "\ats ts", aloak(), workspace.arr);
       esit (255);}
    if ( roods - 0) {
       sprintf(workspace.arr, "Sorry, but to already exists", asse);
       SLOTTES (workspace.arr);
       return (-1);}
    empCmsg();
    return roods; }
 else {
                         /* This is where you land when everything is O.E. */
    wlista();
                                    /* Close the list of choices */
    roods = arollORA(name);
                                    /* Try to get this one added */
    1 ( poode ) {
       sprintf(workspace.arr, "FAILED -- 4s", sqlca.sqlcrm.sqlcrmc);
       fprintf(danta, "\a% % FAILED to earoll", clock(), name);
fprintf(danta, "\a\t\t\t %s", sqlos.sqlorm.sqlormo);
       if (STOORR()) (
          sprintf( workspace.arr, "PENNASES stuck: 4s1",
               sqlos.sqlarm.sqlarmo);
          SLOUTES (workspace.arr);
          fprintf(dante, "\ntsPSNMASHN stuck: SUPERUSER re-connect failed",
                  slock());
          fprintf(dante, "\n\t\t\t *e", sqlca.sqlerrm.sqlerrmc);
          aloseORA():
          amit (255) ; )
       12 (|mlisto()) {
                                    /* Recover by reinitializing screen */
          ubunch () :
          MENVALS(); }
       if ((roods - SQL_RAD_LOGOS) || (roods - NO IDESTIFIED BY)) {
          sprintf (workspace.arr,
             "FAILED -- Are there perhaps spaces in \"ts\"?", meme);}
       SLOUTES (workspace.arr);
    alsa {
                                  /* arollORA() executed O.E. Hext */
       if (fintise()) return 0; /* create the ORACLE tables needed */
       #1fdef CENC
         printf("\nSomething major is wrong. Code = %d", roode);
       Bendie
       SLOUTEF ("Initialization terminated abnormally");
       SLOUTP (sqlos.sqlerm.sqlerma);
       SLOUTP ("This should be fixed before entering data");
       Sprintf (dante, "\nts Initialization did not complete normally", clock());
       fprintf(dante, " due to\n\t\t\t *s", sqlca.sqlerms.sqlerms);
      fprintf(danta, "a\t\t\t This problem should be fixed before proceeding");
fprintf(danta, " with\n\t\t\t data entry for %s", asso);
}
ist peprobet()
      peprobet
                       Routine to set up the ASAN Status Screen. This
                        displays the ASSESSMENT structure. If SUPERUSER
                        is the current user, the system loads the last
                        active assessment first.
*************************
int closeORA(), lastsess(), loggedon(), roods, statusflag;
Char #clock();
#1fdef CEBCEOUT
printf("\npsprobst ");
Sendif.
loggedon();
if (!stromp(ASSESSMONT.name, "SUPERUSER") ) ( /* Not COMMENTED ?
   statusflag = lastsess();
                                               /* Get last session
   if (statusflag — 0) (
                                               /* Found it!
```

```
MANG SQL SELECT REGERES
                                         /* Find the same of the */
             FROM systemilist
                                          /* last assessment
                                          /* worked on
              WHERE weerid = :weerno:
     #1fdef CEBCROOT
        printf(" %ld", sqlos.sqloods);
        die.
     if (sqlos.sqloods - SQL_ROF) ( /* Trouble in River City.... */
        SLOTTER ("PERFORST detected logbook violation. ID = %d", usermo);
        Sprintf(dante, "\nts PSPROBST detected logbook violation. ID = $d",
               clock(), weerno);
        closeORA():
        exit (255) ; }
     roods = ASANoona (a2bv.arr); /* Sign on as that last assessment */
        SLOUISS ("ASANOORS error not trapped");
        MINOCHAG:
        SLOUIF (sqlos.sqlorm.sqlorme);
     return roods; /* ASANoona will have put up the proper screen */
     if (statusflag - SQL_BOF) (
        SLOUTER ("No work ever done yet: You can only start a new one");
        blakdspl();
        MEN SCREEN ("chggurasscreen");
        return 0;
     exposes ();
     return statusflag;
           /* When you are already connected to ASAN as a regular user */
           /* You YOURSELF are now the last user and the time is NOW! */
   stropy(plannist.arr, plannam.arr);
   EXEC SQL SELECT TO_CEAR(SYSDATE, 'dd-Mon-yy EE24:MI:SS')
                FROM dual into :lastdate;
   MENVALS();
   MEN_SCREEN("probetatecreen");
   return 0;
   )
3
ist blakdspl()
blakdspl
                -- Routine to blank out the common display area.
                     This is a safety precention, since if the list
                     to be displayed is empty, whatever was left from
                     the previous list would show. This is because
                    the "bunch" routines don't "remember" how often
                     they have been called and so don't blank out a
                    display when they run out of data!
register int m. 1;
m = (sizeof dsplmult) / 34;
for (1 = 0; 1 < m; 1++) {
    dsplmult[i].arr[0] = '\0';
  deplault[1].lea
                   - 0;}
MENVALS ();
screensm.pc -- Routines to set up the screens for MTMs.
     This file contains:
     pastrent
              - Puts up screen to work with MTRs
     pechgatr
               - Puts up screen to select a new METR
     penuntra
              - Puts up screen to define a new mtr
     pomisroq
              - Puts up screen to define mission require
     mintrixt
              - Make a textblock of an MTR's Navigation Point user data *
     estatzpt
              - Starts the MTR entry process
```

```
axtatzpt
                - Store as MCR Mavingation Point and advance entry one step *
               - Insert as MTR Mavigation Point into the detabase
      insetspt
               - Verify existence of aircraft data for MER calculation
      viyeastr
      vfymetr
                - Verify existence of a MFR on the system
                - Cancal the MTR currently pending on the detabase
                - Counit the MTR currently pending on the detabase
      CERCIE access routines:
       1. ...listo() Open Cursor (Opens a "logical file")
2. ...listf() Fetch Cursor (Reads next record from logical file)
...bunch() Fetch Cursor (Reads next bunch from logical file
                       applicable only for multiple choice options)
       4. ...lists() Close Cursor (Closes the Logical file)
     ... = ms* - Homes of all accessible MTRs (Alphabetical order)
     ... = mt* - Segments of a particular MTR
     ... = ma* - All user information for a particular MFR
              * - Asterisk indicates that multiple choice option is
                   supported for this set of routines.
*****
Singlade (stdio.b)
#define SQLCA_STORAGE_CLASS extern /* Switch for header files
MING SQL INCLUDE SQLCA;
MARC SOL REGIN DECLARE SECTION;
                                    /* All SQL declarations are in
                                    /* these header files
EXEC SOL INCLUDE hostwars.h:
EXEC SQL INCLUDE faharris.h;
EXEC SQL END DECLARE SECTION;
finalude "esen.h"
                                    /* Standard ASAN Sender File
pentreet -- Set up routine for "MTR Data Entry Screen"
     Routine finds last MTR known to have been worked on that is "good"
      (i.e. not one that has been abandoned during entry) makes that the
      current noise source and mixed and puts up the screen.
char *clock():
roid expCmsg();
#1fdef CRECROUT
printf("\npostrent ");
.
.
EXEC SQL SELECT label, type, status, descr, orig,
                                                /* The laundry list */
               sched, owner, TO_CHAR(date_pub,'dd-Mon-yy'),
               TO CHAR(timestamp, 'dd-Mon-yy EE24:MT:88')
sources /* the list of accessible sources */
        FROM SOUTOBS
        13670
              :sraid, :sratype, :sradesc, :sraorig,
               :srosched, :sroowner, :sropdate, :sroedate
                                            /* for this name and */
        WHERE label - : lastmir
                                               /* which is an METR */
              AMD type = '1';
#1fdef CEBCROUT
 printf(" %ld owner %d", sqloa.sqloode, sroowner);
Sendit.
if (sqloa.sqloods) (
  1f (sqlos.sqloods == SQL_BOF) {
     sprintf(srodesc.arr, "There are no MTRs active for 4s", ASSESBEET.name);
     srodesc.len = strlen(srodesc.arr);)
  alse (
     ENDOMES;
     Sprintf (dente,
     "\n$s $s\n\t\t\t looking for MTR $s in Source Library for $d",
     clock(), sqlca.sqlerma.sqlermac, a2bv.arr, ASSESMONT.id);
     emponeg();
```

```
return (int) sqlom.sqloode;}
alsa (
   srodesc.arr[srodesc.len] = '\0';
   sroadate.arr[sroadate.lem] = '\0';
sroadate.arr[sroadate.lem] = '\0';
sroadate.arr[sroadate.lem] = '\0';
   sroorig.arr[sroorig.lea] = '\0';
   srcid.arr[srcid.len] = '\0';}
MENVALS();
NEW_SCREEN ("mtratry");
retura (int) sqloa.sqloods;
peahgatr()
/*****************************
           pechgatr -- Set up routine for screen to change to new MER
      Routine (1) opens oursor for fetch of all MTR names known to system
              (2) fetches the first betch into memory
              (3) puts up "chgounstreorees"
int blakdspl(), mslisto(), msbunch(), roods;
#1fdef CENCEDUT
printf("\mpschgmtr ");
Seediff.
blakdepl();
HEW_SCREEN ("chgourstrscreen");
SLOUT("Retrieving ASAN's list of accessible METHS");
roode = melisto();
return (roode ? roode : mebunch());
int penuntra (neme)
          peaustrn -- Set up routine for new MFR entry screen
     Routine (1) verifies that the new name is unique. If not, the
                 reason (whether it is already on your list or enother assessment already has exclusive use of it) is shown
              (2) puts up the MFR definition screen and starts it
********************************
char zemo[];
char *clock();
int roods, melists(), mebunch();
void exponeg();
#1fdef CENCEOUT
printf("\npeaumtra ");
Seedif.
REMOVE_WINDOW();
SLOUZ("Fromesing New MIR Request");
if (( roods = vfyMatr(name)) != SQL_ROF) { /* Check for strange things */
   1f ( roods = 0) {
     sprintf(workspace.arr, "Sorry, but %s already exists", name);
      ELOUTEP (workspace.arr);
     return (-1);}
  if ( roods - DUFLICATE OBJECT) (
EXEC SQL SELECT Username
```

```
FROM systematist
               2270
                     :tid
               WHERE userid = :sroomer;
      tid.arr[tid.len] = '\0';
      sprintf (workspace.arr,
         "Sorry! to already \"owns\" ts", tid.arr, name);
      SLOUTEF (Workspace.arr);
      return (-1);}
   amponeg();
   fprintf(dante, "\nts Pfneumtr: ts", clock(), sqlca.sqlerm.sqlermc);
   erostat = 'U';
   return reces; }
else {
   melista();
                                              /* Close list of aboless */
   stropy (sroid.arr, a2bv.arr);
                                              /* This is now the mtr name */
   srcid.len = m2bv.len;
   EDEC SQL SELECT TO CHAR(STEDATH, 'dd-Mos-yyyy')

FROM dual IMFO :sropdate: /* Suggest a data of Fubl. */
   sropdate.arr[sropdate.lea] = '\0';
   #1fdef CEBCEOUT
      printf("\mDATE (sqloods = %ld) %n %s", sqloa.sqloods,
                     sropdete.lee, sropdete.err);
   srodesc.len = 0;
   srodesc.arr[0] = '\0';
   srostat = 'A':
   MEN SCREEN ("mtrdefinescreen");
                                             /* Fut up data entry screen */
   MENVALS ();
   return (int) 0; }
int penisreq()
      pemisreq -- Routine to set up the Mission Requirements Son
int HEW SCREEN();
#1fdef CENCHOUT
printf("\npemisreq ");
Seedil.
NEW_SCREEN ("missroq");
return 0;
1
int mkmtrtxt()
mintrixt -- Create a textblock of an MTR's Navigation Points
                   user information (Identifier, Fix info, etc.)
char *clock();
FILE *fopen();
ist i, j, mtlisto(), mtlistf(), mtlista(), falose();
#1fdef CENCHOUT
printf("\maketrtst ");
fendif
if ((txtblkf = fopes("txtblk\\str.txt", "w") ) == NULL) {
  SLOUTER ("Error opening file for help window");
fprintf(dante, "\ats could not open MTR textblock file for $d",
          alock(), ASSESSMENT.1d);
  return (int) DUFLICATE OBJECT; }
fprintf(txtblkf, " Mavigation Points for %s are:\n\n", sreid.arr);
mtlisto();
if (!sqlca.sqlcode) {
  for (;;) {
     mtlistf();
     if (sqlqa.sqlqode) break:
     for (imercid.len, j = 0; i < n2bv.len; i++, j++)
```

```
presexpt.err[j] = a2bv.arr[i];
            fprintf(txtblkf, *4-4s4-5s 403d/403d 4-12s 402d/402d\n*,
                pressypt.arr, prefixed.arr, prefixed, prefixed,
                 prefixtyp.arr, provideft, providright);
     if (sqlos.sqloods t= SQL_MOF) {
        Sprintf (deste,
            "\nes &s\n\t\t looking for MTR segments in MTREGERITS for &d",
     clock(), sqloa.sqlorms.sqlorms, a2bv.arr, Assistantr.id);}
Sprintf(txtblkf,"\a == ND ==\a\a\a\a\a\a\0");
                               - - -\a\a\a\a\a\o*);
     mtliste();
  falose (tatblif);
  return (int) sqlos.sqloods;
  int entmtrot()
  strpt -- Inserts the originating and scheduling estivity in
                       the SOUNCELIST and pope up the window for entering
                       METR mevigation points.
  ************************************
  char *clock();
 #1fdef CEBCHOUT
 printf("\nestmtxpt ");
 EXEC SQL SELECT TO_CHAR(SYNDAME, 'dd-Mon-yy RE24:MI:88')
                     FROM dual INTO :sroadsta;
                                                           /# Bow #/
 #1fdef CERCEOUT
 printf(" DATE = %ld ", sqloa.sqloode);
 -
 srodesc.len = strlen(srodesc.arr); /* While the interface knows, */
sroorig.len = strlen(sroorig.arr); /* ORACLE doesn't that know yet! */
srosched.len = strlen(srosched.arr);
 if (stromp (ASSESSMENT.neme, "SUPERUSER")) (
    SPOOMER - ASSESSMENT.14;
    RING SQL INSERT ISTO BYSOURCES (LAREL, TYPE, STATUS,
                         DESCR, ORIG, SCHED, OWNER, DATE FUB, THESTAMP)
             VALUES (:sreid, /* Name given to this MTR
                        '1',
                                       /* Code identifying MTRs
                                      /* Obsolescence/Applicability
/* English Description
                      :erostat,
                      :srodesa,
                      :sroomig,
                                      /* Originating Activity
                                      /* Scheduling Activity
                      : sroowner,
                                      /* This assessment's id
                     TO DATE(:sropdate,'dd-Mon-yyyy'), /* Date of publ.
                     TO DATE(:sroadsta,'dd-Mon-yy EE24:MI:SS'));}
   EXEC SQL IMMERT INTO noise sources (LAMEL, TYPE, STATOS,
DESC, CRIG, SCHED, OWNER, DATE FUR, TIMESTAMS)
             VALUES (:sroid,
                                   /* Home given to this MTR
                       111.
                                      /* Code identifying MITTER
                                      /* Obsolenscence/Applicability
                      :srostat,
                     :srodesc,
                                      /* English Description
                      :ercorig,
                                      /* Originating Activity
                                      /* Scheduling Activity
                     :sresched,
                     MULI,,
                                      /# SUPERUSER doesn't mess around!
                     TO_DATE(:sropdate,'dd-Mon-yyyy'), /* Date of publ.
TO_DATE(:srondate,'dd-Mon-yy E224:MX:88'));}
#1fdef CEECHOUT
printf(" Insert = %ld", sqlca.sqlcode);
if (sqloa.sqloods) {
   sprintf(workspace.arr, "FAILED &s", sqlca.sqlarms.sqlarms);
   SLOTES (Everkspace.arr[7]); /* Track arrors other than data conversions */
if ((sqloa.sqloods < -1899) || (sqloa.sqloods > -1800))
      fprintf (dente.
```

```
alock(), workspace.arr, srcid.arr, srcowner);
    srostat = '0';
                        /* Leave a special code to prevent committing MTR! */}
 else {
    grostat = '1';
                        /* (Still) looks like a good MER to me! */
    NEW SCREEN ("definodatrscreen");
    presevpt.arr[0] = curnevpt.arr[1] = '\0'; /* Initialize variables */
    premaypt.len
                      = curnavpt.lea
                                          - 0;
    prefixid.arr[0] = curfixid.arr[0] = '\0';
                      = ourfixed.les
                                          - 0;
    profixed.les
                     - curfixred
    prefizred
                                          = 0;
    profindist
                     m curfindist
    prefixtyp.arr[0] = curfixtyp.arr[0] = '\0';
    carmenht.siz.[0] = 'F.' :
strobh(epon:Tet' . 'T.'.', "H.);
    stropy(show.loa, "...x...'..\"W");
    est.lat[0]
                     = eat.lon[0]
                                          = '\0';
    stropy(prehighelt.spec, ".... MML");
    STRCEY (prelowelt.spec, ".... AGL");
                     - ourwidleft
                                         - 0:
    providright
                    - cerwidright
    presitod.arr[0] = curartod.arr[0] = '\0';
    HEWVALS ();
    lptr1 = &curlowalt.altitude; /* Set pointers for ORMILE IMMERTS later */
    lptr2 = &curhighalt.altitude;
    cotr1 = curlowalt.maits:
    optr2 = ourhighalt.maits;
    optr3 = est.let;
    optr4 = est.los;
    dptrl = &est.eastings;
    dptr2 = Sent.northings;
    dptr3 = &est.letitude;
    dptr4 = 6est.longitude;
return (int) sqloa.sqloode;
ist satutspt()
astutrpt -- Store as MFR Mavigation Point and advance the
                    current MAV FOIRT to be the previous.
register int i:
ist insetrpt();
Fifder CERCEOUT
printf("\maxtmtrpt ");
Beadif
if (!iamtrpt()){
    /* Advance the New Point Parameters */
  for (i = 0; i < 3; i++) premavpt.arr[i]
                                               = curnavpt.arr[1];
  for (i = 0; i < 3; i++) preartoc.arr[i]
for (i = 0; i < 5; i++) prefixid.arr[i]
                                              = curartoc.arr[1];
                                               = curfixid.arr[1];
  for (i = 0; i < 10; i++) prelowalt.spec[i] = curlowalt.spec[i];
  for (i = 0; i < 10; i++) prehighalt.spec[i] = curhighalt.spec[i];
  for (i = 0; i < 12; i++) prefixtyp.arr[i] = curfixtyp.arr[i];
for (i = 0; i < 14; i++) show.lat[i] = ant.lat[i];
  for (i = 0; i < 14; i++) show.lat[i]
for (i = 0; i < 14; i++) show.lon[i]
                                              = est.lon[1];
  prefixred
  prefindist = curfindist;
prewidleft = curwidleft;
prewidright = curwidright;
   /* Blank out the New Point Parameters */
  curnavpt.arr[0]
                     - '\0':
                     - '\0';
  est.let[0]
                     = '\0';
  est . los [0]
```

"\m\$s \$s\m\t\t\t entering MTR \$s into Source Library for \$d",

```
est.lstitude
                      - OFF_EARTE;
   est.losgitude
                      - OFT_EARTE;
                      - 0;
   aurflered
                         o ;
   corfied et
   MENVALS ();
return (int) sqlos.sqloods;
}
ist issutrpt()
insutrpt -- Insert as MTR Wavigstics Point into the detabase
int cancetr();
register int 1:
void amoung():
#1fdef CENCROUT
printf("\minestrpt ");
-
if (COOR2utm(Sent)) return (int) -1:
stropy (h2bv.arr, sroid.arr);
stroet (a2bv.arr, surnavpt.arr);
n2bv.len = strlen(n2bv.arr);
EXEC SOL INSERT DIFFO MURRICAGERES
                       (FIX LABEL,
                                      FLOOR_REF, CRILING_REF, FIX_ID,
                                      ARTOC, FIX LAT,
FIX DIST, FLOOR,
                        FIX TIPE,
                                      ARTCC,
                                                              FIX LOSS.
                        FIX RAD,
                        WIDTE LEFT,
                                      WIDTE RIGHT)
                        (:a2bv, :optr1, :optr2, :oursistyp, :oursitoc, :optr3,
                 VALUES (: n2bv.
                                                              : carfixid,
                                                              :optr4,
                        :curfixed, :curfixdist, :lptrl,
                                                              :lptr2,
                        : ourwidleft, : ourwidright);
#1fdef GEBGEOUT
 printf(" Insert1 = %ld", sqlom.sqloode);
  41.5
if (sqlos.sqloods) (
if (sqlos.sqloods - EXISTS) SLOUTES ("Duplicate Navpoint Id");
   else empCmsg();
   omnumetr();} /* This is necessary since ORACLE automatically rolls back */
               /* the inserts already done at this point ! */
also (
   EXEC SQL INSERT INTO MAVFOIRTS ( FIX LARGE, X, Y, LAT, LOW )
                       VALUES( :n2bv, :dptr1, :dptr2, :dptr3, :dptr4);
     printf(" Insert2 = %ld", sqlom.sqloods);
   Beadle
   if (sqlca.sqlcode) {
     emplessy ();
      if (sqlos.sqlwarn[0]) expourn();
      cancestr();} /* ORACLE automatically rolls back */
                  /* inserts after this occurs 1
return (int) sqloa.sqloode;
}
int vivacetr(ness)
vfysomtr -- Verify the existence of mircraft data for
                        MFR calculations in the detabase
     Routine looks in MTRSELFAR (a view on a READQUARTERS table), loads
     pr pwr u with the power units for which we have data for the aircraft and returns sqlcm.sqlcode for the query. Sero means MTR exists, SQL_ROF means it does not, all other values indicate an SQL error.
```

```
cher name []:
 register int 1;
  char *clock();
 Bifdef CHRCHOUT
 printf("\avfyecetr ");
 .
Jesaif
 m2bv.lem = strlem(meme);
 for ( i=0; i < a2bv.len; i++)
    n2bv.arr[1] = nama[1] = toupper(nama[1]);
 Pr_Pwr_m.arr[0] = '\0';
 pr_pwr_u.lea = 0;
                                     /* See if aircraft power units */
/* are in the list of aircraft */
 EXEC SQL SELECT power_waits
         FROM streetab
INTO :pr_pwr_u
WERNE miroraft = :m2bv;
                                     /* for which we have MTR data */
 Sifdef CENCEDUT
   printf(" tld to power in to", sqlom.sqloods, n2bv.arr, pr_pur_m.arr);
 if (sqloa.sqloods) {
   if (sqloa.sqloods === SQL_BOF) {
      sprintf(workspace.arr, "META Calculation not supported for %s", a2bv.arr);
}
       SLOUISF (workspace.arr);}
    else {
       Sprintf (deste,
       "\ats ts\a\t\t\t looking for ts in MTREELING for td",
      clock(), sqlca.sqlerm.sqlermq, a2bv.arr, ASSESMENT.id);}
 else {
   pr_per_u.arr[pr_per_u.lea] = '\0';
for ( i=0; i < ntbv.lea; i++)
      sc_neme.arr[1] = a2bv.arr[1];
   as arms.les w 1:
   ac_asso.arr[i] = '\0';
    EXEC SQL SELECT power, speed
                                 /* Load the reference conditions */
/* as initialization for this */
         FROM mtreeltab
         2000 :so_our_pwr, :so_our_spd
                                              /* Aircraft/Mission
         WEEKE alrereft = :ac aumo;
   ac_pre_pwr = ac_cur_pwr;
   ac_pre_spd = ac_our_spd;
 #1fdef CERCEOUT
  printf(" tld to power in to", sqlca.sqlcode, a2bv.arr, pr_pwr_u.arr);
Beadle
   MENVALS(); }
return (int) sqloa.sqloode;
wfyMstr -- Verify the existence of a MFR on the system
      Routine looks in SOURCELIST (a view on a SUPERUSER table), loads
      sroomer and returns sqlos.sqloods for the query. Sero meens MFR
      exists, SQL BOT means it does not, all other values indicate an SQL
      error except that DUPLICATE CRUECT means that the object exists on
      the system but is not accessible to the present assessment
char nemo[];
register int 1:
#1fdef CEECEOUT
printf("\nwfyMatr ");
feedif
```

```
n2bv.len = strlen(neme);
for ( i=0; i < m2bv.lem; i++)
    a2bv.arr[i] = toupper(neme[i]);
MERC SQL SELECT OWNER
                                  /* Identify the owner from */
/* the list of all sources */
        FROM sourcelist
        NEEDE label = :m2bv
                                  /* which has this name and */
              AND type = '1';
                                  /* is an MER.
#1fdef CEBCEDOT
  printf(" %ld owner %d", sqloa.sqloods, sroowner);
if (sqlos.sqloods - EULL_FETCHED) return (int) 0;
if ((sqlom.sqloods === 0) 44 (sroomer != ASSESSMENT.id))
return (int) DUPLICATE OBJECT;
return (int) sqloa.sqloods;
ist concert:()
/**********************************
       concentr -- Concel the META currently pending on the detabase
     Routine always issues a ROLLANCE
ist postreet();
#ifdef CHBCROUT
printf("\momenantr ");
feedif
EXEC SQL ROLLBACK WORK:
Fifdet CERCHOUT
  printf("tld", sqloa.sqloode);
sprintf (workspace.arr, "Entry for MER to CAMCELLED", sreid.arr);
SLOUTS (workspace.arr);
srcid.arr[srcid.len = 0] = '\0';
postreat();
return (int) sqlos.sqloods;
savestr -- Commit the MTR currently pending on the database
     Routine checks if there is a valid MTR being constructed them
     issues a COMMIT to the detabase
*****************************
register ist 1;
#1fdef CENCEOUT
printf("\nsavemtr ");
-
mtlista();
if (srostat - '0') (
  SLOUTS ("MIR had errors and was not saved");
  MICHE SQL ROLLBACK WORK;
  #1fdef CERCEOUT
    printf("ROLLED MER BACK!.... %ld", sqlos.sqloods);
  Sendif
  srcid.arr[srcid.lea = 0] = '\0';
alse { /* First check if we have to enter one more point: */
  if (ournewpt.arr[0] |= '\0')
```

```
if (insutrpt()) return (int) sqloa.sqloods;
   MORE SOL COMMET WORK:
   #1fdef CENCEOUT
      printf("Committed MTR %ld", sqloa.sqloods);
   Seedif.
   street():
 return (int) sqlos.sqloods;
 int melisto()
 melisto --- Open cursor $1 for a list of MITS on the system
                   that are accessible to an assessment
     Routine executes an open cursor command for cursor $1 and then
     returns to the celling progrem with the CRACIAL status code.
     Note: Modifications to this function may impact the related
            functions melistf(), mebunch() and melisto() that fetch
            rows and close the cursor and, possibly, functions that
            call these utility routines.
 #1fdef CHBCROOT
 printf("\mmslisto ");
 -
 MXBC SQL DECLARS S1 CURSOR FOR
        SELECT label
                              /* These are unique identifiers
        FROM sources
WEERE type = '1'
                              /* View of all accessible "sources" */
                              /* MTRs are type - 1
        ORDER BY label;
 EXEC SOL OPER S1:
 #1fdef CEBCEDOT
 printf(" Open: %ld ", sqloa.sqloods);
 Sendi f
 return (int) sqlom.sqloods;
int melistf()
          melistf --- Fetch a row using the opened cursor $1 for MTMs
     Routine executes an fetch command for cursor $1, which is assumed
     to have been opened, and them returns to the calling progrem with
     the CRACLE status code.
     Note: Modifications to this function may impact the related
            functions melisto(), mebunch() and melisto() that open,
            fetch groupwise and close the cursor and, likely,
            functions that call these utility programs
 EXEC SQL FETCE S1 1870 :sroid;
sraid.arr[sraid.lea] = '\0';
#1.fdef CHRCHOTT
printf(" Fetch: %ld ", sqlca.sqlcode);
Seedif.
retura (int) sqlog.sqloode;
int mebunch()
--- Fetch a bunch (20 or whatever the size of deplault)
                 using the opened cursor $1 for strlist
```

```
Routine executes an fetch command for cursor $1, which is assu-
      to have been opened, and them returns to the calling program with
      the CRACIAL status code.
     Note: Modifications to this function may impact the related
functions melisto(), melistf() and melisto() that open,
             fetch and close the cursor and, likely, functions that
             call these utility programs
register ist i:
int arows, arows, reads;
char *clock();
NXEC SQL FETCE S1 INTO :deplmult;
arows = (int) sqlox.sqlerrd[2];
mrows = (sizeof deplett) / 34;
roode = (int) sqlos.sqloode;
#1fdef CENCEOUT
printf(" meBunch: $1d returns $4 of $4 rows", sqlom.sqloods, arows, arows);
 feedic
if ((roods - SQL_FETCH_OUT_OF_ONDER) || (roods - SQL_BOF)) {
   if ((reeds - SQL_BOF) && (arows > 0)) {
      for (i = 0; i < aross; i++) deplault[i].arr[deplault[i].lea] = '\0';
      if (arows < mrows)
        for (i = arows; i < arows; i++) deplant[i].arr[0] = '\0';
      HENVALS ();
      SLOOTS ("The last MER in the list is on the sursen"); )
   alsa (
     HENVALS ();
     SLOTTP ("You are already as far down in the list as you can go"); }
alse {
  if (roods) {
     HINDOMS;
     fprintf(dante, "\ats MSEUSCE: ts", clock(), sqlca.sqlerm.sqlermc);}
  MENVALS ();}
return roods:
int melista()
--- Close Gursor Si for mirlist
     Montine executes a close cursor command for cursor $1 and then
     returns to the calling program with the CRACLE status code.
     Note: Modifications to this function may impact the related
            functions melisto(), mebunch, and melistf() that open the
            cursor and fetch rows using it and, possibly, functions
            that call these utilities
*************************************
#1fdef CHRCHOUT
printf("\mmslistc ");
Sendid.
EXCEC SQL CLOSE S1;
printf("Close: %ld ", sqlos.sqloods);
Sendif
retura (int) sqloa.sqloode;
int mtlisto()
mtlisto --- Open oursor 82 for a list of user information
                    of navigation points on a particular MTR
     Routine executes an open cursor command for cursor $2 and then
```

```
returns to the calling program with the CRACIES status code.
      Note: Modifications to this function may impact the related
             functions mtlistf() and mtlistc() that fetch rows and
             close the cursor and, possibly, functions that call
             these utility routines.
 #1fdef CENCROTT
 printf("\mmtlisto ");
 -
 stropy( aid.arr, sraid.arr);
 stroat (old.arr, "4");
 oid.len = strlen(oid.arr);
 EXEC SQL DECLARE S2 CURSOR FOR
         SELECT fix_label, fix_id,
                                    fiz_type,
                                               fiz_red,
                fix_dist, width_left, width_right
        FROM strangments
WHERE fix_label LIKE : aid
         OFFER BY fiz_label;
EXEC SQL OPEN S2;
printf(" Open: %ld ", sqloa.sqloods);
 Bendif
 return (int) sqlos.sqloods;
int mtlistf()
/*********************************
     milistf --- Notch a row using the opened cursor $2 for MTR
                    Mavigation Point User Information
     Routine executes as fetch command for oursor $2, which is assu-
     to have been opened, and them returns to the calling program with
     the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions mtlisto() and mtlisto() that open and close the
cursor and, likely, functions that call these mtilities
**********************************
EXEC SQL FERGE S2 INTO :a2bv, :prefixed, :prefixtyp, :prefixred,
                    :prefindist, :prewidleft, :prewidright;
prefixid.arr[prefixid.lea] = '\0';
prefixtyp.arr[prefixtyp.lea] = '\0';
#ifdef CERCEDUT
printf(" Fetch: %ld ", sqlca.sqlcode);
-
retura (int) sqloa.sqlooda;
          *************
          mtlists --- Close cursor $2 for Mavigation Points
     Routine executes a close cursor ocumend for cursor $2 and then
     returns to the calling program with the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions mtlisto() and mtlistf() that open the cursor
            and fetch rows using it and, possibly, functions that
            call these utilities
```

```
#1fdef CEECHOOT
printf("\matlista ");
 .
Seedi £
EXEC SOL CLOSE S2:
Bifdef CENCEOUT
printf("Close: %ld ", sqlos.sqloods);
feedif.
return (int) sqlos.sqloods;
int malisto()
      malisto --- Open oursor $3 for a list of user information
                      of mavigation points on a particular MER
     Routine executes an open cursor command for cursor $3 and then
      returns to the calling program with the CRACIE status code.
     Note: Modifications to this function may impact the related
              functions mtlistf() and mtlistc() that fetch rows and
              close the cursor and, possibly, functions that call
              these utility routines.
#1fdef CENCHOUT
printf("\mmalisto ");
.
Jendif
stropy( cid.arr, srcid.arr);
stroat (old.arr, "4");
mid.len = strlen(mid.arr);
EXEC SQL DECLARE SS CURSOR FOR
        SHIECT fix label, floor ref, calling ref, fix id, fix type, artor, fix lat, fix loa, fix red, fix dist, floor, calling, width left, width right
        FROM
                 mtragments
               fix label LINE : cid
        ORDER BY fix label;
EXEC SOL OPER 43:
#1fdef CHECKOUT
printf(" Open: %ld ", sqloa.sqloods);
retura (int) sqlca.sqlcode;
int malistf()
malistf --- Fetch a row using the opened cursor 83 for MTR
                     Mavigation Point User Information
     Nottine executes an fetch command for cursor #3, which is assumed
     to have been opened, and then returns to the calling program with
     the ORACLE status code.
     Note: Modifications to this function may impact the related
             functions mtlisto() and mtlisto() that open and close the
             cursor and, likely, functions that call these utilities
*); /* If you don't use VARCHAR, you have to */
*); /* clear the space or waird things happen */
stropy(optr3, "
stropy(optr4,"
EXEC SQL PETCE S3 THTO : m2bv.
                                  :optrl,
                                               :optr2,
                                                           : corfixed,
                      courfixtyp, courartos, coptr3,
courfixed, courfixdist, clptr1,
convoidingt, courwidingts;
                                                           :optr4,
                                                           :lptr2,
```

```
prefixed.arr[prefixed.lea] = '\0';
profixtyp.arr[profixtyp.lea] = '\0';
#1fdef CERCHOUT
printf(" Fetch: %ld ", sqloa.sqloods);
-
return (int) sqloa.sqloada;
ist malists()
malista --- Close cursor $3 for Mavigation Points
     Routine executes a close cursor command for cursor $3 and them
     returns to the calling progrem with the CRACLE status code.
     Note: Modifications to this function may impact the related
            functions milisto() and milists() that open the cursor and setch rows using it and, possibly, functions that
            call these stilities
#1fdef CENCROOT
printf("\mmtlista ");
Bearle F
ECHEC SQL CLOSE S3;
printf(" Close: %ld ", sqlos.sqloode);
Sendif.
return (int) sqlos.sqloods;
/***********************
     startup.pd -- ASAN'Initialization Code. This program is only
                   used to determine the status of ORACLE and to
                   load those pieces which are needed. The space
                   it occupies can be relimquished after execution.
/* The usual stuff, of course */
/* Header for calls to MS-DOS */
#include <stdio.h>
#include 
process.h>
                             /* String manipulation header */
#include <string.b>
Singlade <time.b>
#define SQLCA STORAGE CLASS extern
EXEC SQL BEGIN DECLARS SECTION;
EXEC SQL INCLUDE hostvars.h;
EXEC SQL INCLUDE faharris.h;
EXEC SQL END DECLARE SECTION;
EXEC SQL INCLUDE SQLCA;
Singlade "agen.h"
strtASAN()
/**********************************
     strtASAN -- start ASAN function
        Starts by attempting to connect to ORACLE as "SUPERCER".
        If ORACLE is not up it will attempt to install it. If
        unsuccessful, execution will terminate with appropriate
        diagnostic information (i.e. what piece of the ORACLE
        detabase manager it cannot find) .
    Note: This program is totally dependent on how ORACLE and its
          associated programs behave. It should be tested (with
          #define CEECEOUT here and in other files with ORACLE calls)
          when a new ORACLE release is installed. Particularly the
          return codes for uninstalled systems, since they are not
```

```
sording to specification in release 5.1. (e.g., -3120
            should be -3121).
*******************************
ist roode, zoode:
                                   /* Oracle return code temporaries */
int doods = 0;
                                   /* MM-DOS return code */
int spewip(), Stooms();
cher *clock();
stropy (Assessment . name, "SUPERCEER");
00 roods = #Uccan():
                                  /* Try SUPERUSER and see what happens */
if (roods = 00_roods) {
                                 /* 0 = ORACLE up and running */
   switch (roode) (
                    /* 3120 - SQLERE is not installed */
      doods - spewalp (F_MAIT, "sqlpme.eme", "sqlpme.eme", "/moior", MULL);
      12 ( doods < 0 ) (
        printf("\nsystem not properly installed! ASAW com't find");
         printf("\modRCLE's Protected Mode Executive SQUME.EXE");
         printf("\nYour Data Administrator should be able to help\n\n\n");
         fprintf(dante, "\nts SQLFME Failed (4d)", alook(), doode);
         amit (16);}
      else if ( doods < 768 ) {
        printf("\noracle's Protected Mode Executive SQLPME.EXE");
        printf("\nAbsormally terminated. Return code was 4d", doode);
        printf("\aYour Data Administrator should be able to help");
        fprintf(danta, "\nte SQLFME Failed (%d)", aloak(), doods);
         exit (16);}
     fprintf(dente, "\ate SQLFME Installed", slock());
  case ORA_UMAVAILABLE: /* Not available: Do IOR first */
     doods = spewalp(P_WAIT, "ior.exe", "ior.exe", "warm", MULL);
     1f ( doode < 0 ) {
        printf("\nSystem not properly installed! ASAW cen't find");
        printf("\monacte's Startup routine IOR.EXE");
        printf("\nYour Data Administrator should be able to help");
        Sprintf(dante, "\nts CRACLE Server Failed(%d)", clock(), doode);
        exit (16);}
     else if ( doods > 0 ) {
        printf("\nORACLE's Initialization Routine IOR.EXE has");
        printf("\nAbnormally terminated. Return code was 4d", doods);
        printf("\aYour Data Administrator should be able to help");
        fprintf(dente, "\nts CRACLE Server Failed (%d)", clock(), dcode);
        exit (16);}
                          Give COMMET one more try
                     /*
        fprintf(dante, "\nts ORAGLE Server Started", slock());
        1f ( zoode = 50ccan() ) {
          printf("\nAsam has attempted to install the ORACLE Detabase");
           printf("server, but after what \nappears to have been a ");
           printf("\successful installation ASAN doesn't want to start");
           printf("\nCause: %s", sqlca.sqlarma.sqlarma);
           if (sqlca.sqlcode - sqL_RAD_Locom)
      printf("\s\tASAW installation program has probably not been run yet");
          printf("\nYour Data Administrator should be able to help\n\n");
          system("peuse");
           closeCRA();
          fprintf (dante, "\nes Initial Connect failed second time (td)",
                   aloak(), zoode);
          exit(16);}
 default: (
    Sprintf(dente, "\mbs ASAN cannot establish communication with CRACLE",
            slock());
    fprintf(dante, "\nCause: %s", sqlca.sqlerms.sqlerms);
    printf("\n\nASAW cannot establish communication with the ORAGE ");
printf("Database Server. \nCause: %s", sqlca.sqlerm.sqlermo);
    if (sqlos.sqloods - sqt_NAD_Logos)
   printf("\n\tASAW installation program has probably not been run yet");
printf("\nYour Data Administrator should be able to help\n\n");
    system("pause");
```

```
aloseORA():
     exit (16);}
   } /* End Switch */
   } /* Bad If */
 fprintf(dante, "\nts ASAW Started ", clock());
return 0;
int vey_assu()
VIY_ASAN -- Routine to check ASAN release levels and other system
                 integrity functions. This function is not implemented
in nor meaningful for the prototype version of ASAN
**********************************
int pwaheak()
/****************************
      pweheck -- Boutine to check password and/or user name validity
                 This function is not implemented in the prototype
return (int) 0:
tables.pd -- Set of routines to access the data dictionary's
                   inventory of users, tables, columns etc.
    For each ORACLE cursor there are three routines as follows:
      1. ...listo() Open Cursor (Opens a "logical file") *
2. ...listf() Fetch Cursor (Reads next record from logical file) *
         ...bunch() Fetch Cursor (Reads next bunch from logical file
                    applicable only for multiple choice options)
       4. ...listc() Close Cursor (Closes the logical file)
    Routines in this file:
     ... * * * - Emmes of assessments (LIFO order on data started).

t - Names and comments of all tables CREATED by current
                assessment only.
           c - Mamos and com
                             ents of columns in specific table [:tid]
                 (retrieved in order of column in the table).
           to - Homes and comments of columns in all tables CHEATED by
                 current assessment (sorted alphabetically on table and
                 column within table)
           v - ASAN table_of_contents in alphabetic order.
            * - Asterisk indicates that multiple choice option is
                supported for this set of routines.
finclude <stdio.b>
                            /* The usual stuff, of course */
                            /* Ecader for calls to MS-DOS */
/* String manipulation header */
infine SQLCA_STORAGE_CLASS extern
EXCEC SQL INCLODE SQLCA;
                            /* SQL Communication Area
EXEC SOL REGIS DECLARS SECTION:
EXEC SOL INCLUDE hostware.h:
EXEC SQL INCLODE faharris.h;
EXEC SQL END DECLARE SECTION;
#include "assa.h"
                            /* Standard ASAW Header file */
ist ulisto()
```

. .....

```
ulisto --- Open cursor V1 for userlist (ASAN assessments)
      Boutine executes an open cursor command for cursor VI and then
      returns to the calling program with the CRACKE status code.
      Note: Modifications to this function may impact the related
             functions wlistf(), wheath() and wliste() that fotch rows
             and close the cursor and, possibly, functions that call
             these willity programs
 #1fdef CENCEDUT
 printf("\mulisto ");
 feadle
 EXEC SQL DECLARE VI CURSOR FOR
                   SELECT RESTRICT
                                       /* Assessments are ORACLE Users */
                   FROM sysusarlist,
                                       /* Standard Data Dictionary view */
                       table of contents /* ASAN's List of assessments.
                   WEERE systerlist.userid = table of contests.idnumber
                   ORDER BY timestamp DESC; /* LIFO Listing Order */
 MANUE SQL OFFER UL;
 #1fdef CHBCEOUT
 printf(" Open: %ld ", sqloa.sqloods);
 fendif
 return (int) sqlos.sqloods;
 int whists()
 ulistf --- Fetch a row using the opened cursor Vi for userlist
      Nowtine executes an fetch command for oursor VI, which is assumed
      to have been opened, and them returns to the calling progrem with
      the ORACLE status code.
     Note: Modifications to this function may impact the related
             functions ulisto(), ubunch() and ulisto() that open,
             fatch groupwise and close the cursor and, likely, any
             functions that call these utility programs
************************************
EXEC SQL FETCE VI DETO : wid:
wid.arr[wid.len] = '\0';
#1fdef CEECEOUT
printf(" Fetch: %ld ", sqlca.sqlcode);
-
return (int) sqloa.sqloode;
int ubunch()
--- Fetch a bunch (20 or whatever the size of depletalt)
                   using the opened cursor VI for systemilist
     Routine executes an fetch command for cursor VI, which is assumed to have been opened, and then returns to the calling program with
     the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions ulisto(), ulistf() and ulista() that open and
close the cursor and, likely, functions that call these
            utility programs
****************************
register int i;
int arows, mrows, reode;
char *clock();
```

```
MINIC SOL PETCH UI INTO :deplement:
 arows = (ist) sqlos.sqlerrd[2];
 mrows = (sizeof deplexit) / 34;
 roode = (int) sqlca.sqlcode;
 #1fdef CHRCKOUT
 printf(" Bunch: %ld returns %d of %d rows", sqlos.sqloods, arows, arows);
 -
 if ((roods - SQL_FETCH_OUT_OF_ORDER) || (roods - SQL_ROF)) (
    if ((roods -- SQL_BOF) 66 (arows > 0)) (
      for (i = 0; i < arous; i++) deplault[i].arr[deplault[i].lea] = '\0';
      if (mrows < mrows)
         for (i = arows; i < arows; i++) deplarat[i].arr[0] = '\0';
      HENVALS ();
      SLOUTS ("The last assessment in the list is on the sureen"); }
    alsa {
     MENVALS();
      SLOTTE ("You are already as far down in the list as you can"); }
 else if (roode) fprintf(dante, "\nts UNCONCE: ts",
                     aloak(), sqloa.sqlerma();
 MENVALS():
 return reade:
 mlists --- Close sursor V1 for mearlist
      Routine executes a close cursor command for cursor Ul and then
      returns to the calling program with the ORACLE status code.
      Note: Modifications to this function may impact the related
             functions ulisto(), ulistf() and ubunch() that open the
             cursor and fetch rows using it and, possibly, functions
             that call these utilities
 ******************************
 #1fdef CHRCHOUT
printf("\aulistc ");
 Ben41#
EDEC SOL CLOSE VI:
#1fdef CHBCROOT
printf("Close: %ld ", sqlos.sqloods);
Bendiff
return (int) sqloa.sqloode;
int tlisto()
tlisto
            --- Open cursor W2 for list of tables in an assessm
     Noutine executes an open cursor command for cursor U2 and then
     returns to the calling program with the CRACIE status code.
     Note: Modifications to this function may impact the related
            functions thistf() and thisto() that fetch rows and
            close the cursor respectively and, possibly, functions
            that call these utilities
#1fdef CHBCROOT
printf("\ntlisto");
Acads C
MINE SOL DECLARS U2 CURSOR FOR
              SELECT name, cont
                                /* That is what table name and */
              FROM sys.v4emptab /* comment are called in this view */
              ORDER BY NAME;
MICHE SQL OPEN U2;
```

172

```
#15def CHBCROTT
printf(" Open: %ld ", sqlom.sqloods);
Seedif.
return (int) sqlos.sqloode;
int tlistf()
    thistf --- Fetch a row using the opened cursor W2 for tablelist
    Routine executes an fetch command for cursor U2, which is assumed
     to have been opened, and them returns to the calling program with
    the CRACIE status code.
    Note: Modifications to this function may impact the related
            functions thisto() and thisto() that open and close
            the cursor respectively and, possibly, functions that
            call these willities
EXEC SQL FETCE U2 INTO :tid, :workspace;
tid.arr[tid.len] = '\0';
workspace.arr[workspace.len] = '\0';
Fifdef CEBCEOUT
printf(" Fetch: %ld", sqloa.sqlooda);
Sendif.
return (int) sqlos.sqloods;
ist tlisto()
tlists --- Close cursor V2 for tablelist
    Routine executes a close cursor command for cursor U2 and them
    returns to the calling program with the CRACLE status code.
    Note: Modifications to this function may impact the related
            functions tlisto() and tlistf() that open the cursor and
           fetch rows using it respectively and, possibly, functions
           that call these utilities
#Afdef CHRCHOUT
printf(" tliste ");
EXEC SQL CLOSE U2;
Bifdef CENCROTE
printf("Close: %ld ", sqlos.sqloode);
#endi#
return (int) sqloa.sqloode;
int disto()
           --- Open cursor W3 for list of columns in a table
    Routine executes as open cursor command for cursor V3 and then
    returns to the calling progrem with the ORACLE status code.
    Note: Modifications to this function may impact the related
           functions clistf() and clistc() that fetch rows and
           close the cursor respectively and, possibly, functions
           that call these utilities
```

```
#1fdef CHBCBCCT
printf("\malisto");
Bend1f
MING SQL DECLARE US CURSOR FOR
                                   /* That is what the column is */
              PROM
                    col
                                   /* called in this view
                                   /* For the current table
              WHERE tames w :tid
                                   /* In order of exection
              ORDER BY colmo:
EDESC SOL OFFER US:
#1fdef CERCENTY
printf(" Open: %ld ", sqlom.sqloods);
-
retura (int) sqloa.sqloode;
int clists()
clistf --- Fetch a row using the opened cursor V3 for column list *
     Routine executes an fetch command for cursor U3, which is assumed
     to have been opened, and then returns to the calling program with
     the CRACIE status code.
           Modifications to this function may impact the related
          'functions clisto() and clisto() that open and close
           the cursor respectively and, possibly, functions that
            call these utilities
EXEC SQL FETCE US DITO : aid;
tid.arr[tid.lea] = '\0';
#1fdef CEBCECOTT
printf(" Fetch: %ld", sqloa.sqloode);
feedif
return (int) sqlos.sqloods;
int alista()
clists --- Close cursor U3 for teblalist
     Routine executes a close cursor command for cursor V3 and them
     returns to the calling program with the ORACLE status code.
    Note: Modifications to this function may impact the related functions elisto() and elistf() that open the cursor and
           fetch rows using it respectively and, possibly, functions
           that call these utilities
#1fdef CHRCHOUT
printf(" alista ");
EXEC SQL CLOSE T3;
#1fdef CHECKOUT
printf("Close: %ld ", sqloa.sqloods);
fendif
return (int) sqlcs.sqlcode;
int talisto()
tolisto --- Open oursor U4 for list of all columns sorted on table *
```

```
Nortine executes an open cursor command for cursor U4 and then
      returns to the calling program with the CRACIAE status code.
     Note: Modifications to this function may impact the related
functions tolistf() and tolisto() that fotch rows and
             close the cursor respectively and, possibly, functions
             that call these utilities
#1fdef CENCROUT
printf("\stalisto");
EXEC SOL DECLARS U4 CURSOR FOR
                  SELECT tame.
                                       /* TRUME is the table name */
                                       /* CHANGE that of the column */
                         -
                                       /* in the COL view of the */
                         reserve
                                       /* ORACLE Data Dictionary */
                  FROM col
                  ORDER BY tame. on
MANUEL SOL COMMET TA;
Bifdef CHICKOTT
printf(" Open: %ld ", sqlom.sqloods);
-
return (int) sqlom.sqloods;
int tolistf()
    telistf --- Fetch a row using the opened sursor U4
                   (column names sorted by table name)
     Routine executes an fetch command for cursor U4, which is assumed
to have been opened, and then returns to the calling program with
the ORACLE status code.
     Note: Modifications to this function may impact the related
             functions tolisto() and tolisto() that open and close
             the oursor respectively and, possibly, functions that
             call these willities
EXEC SQL FETCE V4 INTO :tid, :did, :workspace;
tid.arr[tid.len] = '\0';
did.arr[did.len] = '\0';
workspace.arr[workspace.lea] = '\0';
Sifdef CHECKOUT
printf(" Fetch: %ld", sqlca.sqlcode);
Sendif
return (int) sqlom.sqloode;
int talista()
/***********************************
        talists --- Close cursor U4 for column list sorted by table
     Routine executes a close cursor command for cursor V4 and then
     returns to the calling program with the ORACLE status code.
    Note: Modifications to this function may impact the related
            functions tolisto() and tolistf() that open the ourser and
            fetch rows using it respectively and, possibly, functions
            that call these utilities
```

#ifdef CENCEDUR

```
printf(" teliste ");
Bendif
EXCEC SQL CLOSE T4;
#1fdaf CHBCHOUT
printf("Close: %ld ", sqlom.sqloode);
feedig
return (int) sqlom.sqloads;
ist vlisto()
vlisto --- Open cursor W5 for list of ASAM's assessments
     Noutine executes an open cursor command for cursor WS and then
     returns to the calling progrem with the ORAGLE status code.
     Hote: Modifications to this function may impact the related functions vlistf() and vlistc() that fetch rows and close
            the oursor and, possibly, functions that call these
            stility programs
#1fdef CEBCHOOT
printf("\avlisto ");
.
Jendif
MORE SOL DECLARS US CURSOR FOR
                  SELECT username, description
                   FROM sysuserlist,
                        table_of_comtemts
                   WEEKE systeerlist.userid = table_of_contents.idnumber
                ORDER BY RESTRONG DESC:
EXCRC SQL OFFER US;
#1fdef CENCEOUT
printf(" Open: %ld ", sqlom.sqloode);
Seeds f
return (int) sqlos.sqloods;
int wlistf()
     vlistf --- Fetch a row using the opened cursor US for list of
                  ASAN assessments and their descriptions
     Routine executes an fetch command for cursor \overline{u}5, which is assumed to have been opened, and then returns to the calling program with
     the ORACLE status code.
     Note: Modifications to this function may impact the related
            functions vlisto() ulista() that open alose the cursor
            and, likely, any functions that call these utility programs
EXEC SQL FETCE US 1870 : wid, :workspace;
mid.err[mid.lem] = '\0';
workspace.arr[workspace.lea] = '\0';
#1fdef CERCEOUT
printf(" Fetch: %ld ", sqloa.sqloods);
retura (int) sqlos.sqloode;
int vlista()
ulisto --- Close oursor US for ASAM table_of_contents
```

```
Routine executes a close oursor command for oursor US and them
      returns to the calling program with the ORACLE status code.
      Note: Modifications to this function may impact the related
            functions ulisto() and ulistf() open the cursor and fetch
rows using it and, possibly, functions that call these
            wtilities
 ***********************
 Biffed CENCERT
 printf("\avlisto ");
 Feedle
 EXCEC SQL CLOSE US;
 fifdef Chicagor
 printf("Close: 4ld ", sqlos.sqloods);
 Seedif.
 return (int) sqlom.sqloode;
 util.pc -- A few general utilities
     This file contains:
                 - displays expansion of ORACLE error message
- displays expansion of ORACLE warning message
      embons.
                 - returns time/date information
 fincinde oprocess.h>
                               /* Header for calls to MM-DOS
 finciade (stdio.b)
 #define SQLCA_STORAGE_CLASS extern
                               /* Switch for header files
 EXEC SQL BEGIN DECLARE SECTION;
                               /* All SQL declarations for this */
 EXEC SQL INCLUDE hostvars.h;
                               /* are in these two header files */
/* this one comes from "U" */
 EXEC SQL INCLUDE Sabarris.h;
EXEC SQL END DECLARS SECTION;
 MERIC SQL INCLUDE SQLCA;
 finclade "asan.h"
                               /* Standard ASAN Reader File
                                                           */
Told exposes ()
ampOmeg -- Expand Oracle Error Massage
fifdef CENCEDOR
  printf("\nt.70s\n", sqloa.sqlerma.sqlerma);
  IDEDCMESS:
  SLOUTS (sqlcs.sqlerms.sqlerms);
Seed12
woid expower ()
expowers -- Expend Oracle Warning Message
***********************
if (sqlca.sqlwarn[1] == 'W')
        SLOUTER ("SQLMANNING: Column Truncated");
if (sqlos.sqlwarm[2] - 'W')
       SLOUTER ("SQLWARSTENG: Hull in aggregate");
if (sqlos.sqlwara[3] = 'W')
       SLOUTER ("SQLWARRING: INTO var sount != col count");
```

```
if (sqlom.sqlwarm[4] -- 'W')
         SLOUTED ("SQLWENTING: Update or Delete w/o WEERE");
 if (sqloa.sqlwara[5] - 'W')
         SLOUTER ("SQLMARSTERG: ?????");
 if (sqloa.sqlwarn[6] - 'W')
         SLOUTER ("SQLWARHING: Rollback Required");
 if (sqloa.sqlwarm[7] == 'W')
         SLOUZER ("SQLWARWING: Change after query for UPDATE");
1
 char *elock()
 slock -- Routine that returns the location of the string
                   with the date and time of this very moment
long now, time();
cher *ctime(), *text;
#1fdef CENCEDOT
   printf("\malock ");
Bendif
now = time(MULL);
text = ctime (show);
text[24] = ' ';
return test;
VCAPITAL (x)
         WCAPITAL -- Convert a WARCHAR entered by user to uppercase
****************************
VARCEAR *=;
register int 1, j;
j = x->lea = strlea(x->err);
for (i=0; i<j; i++) x->err[i] = toupper(x->err[i]);
     These are a set of dummy routines while the program is under
     development. They will be removed in the production version.
-----()
SLOUTS ("This feature is not available in this prototype version");
dumyt ()
SLOOPS ("This feature currently runs as a separate program.");
/************************************
   utmoonw.pg -- Routines dealing with the wonderful world of UTM,
                 DLG, GRASS and such
* This file contains:
  1. wim to lat/lon conversion (CC_u2ll) routines
2. lat/lon to utm conversion (CC_ll2u) routines
     OC_u2ll_spheroid (spheroid_name) must be called first, sets the
             heroid parameters for the ellipse 'spheroid name' (see
```

```
get_spheroid.s for known spheroids). Function is used for
               conversions in either direction.
        CC_u2ll_spheroid_permeters (a,e) called by CC_u2ll_spheroid() to
set the ellipsoid major axis 'a' and eccentricity squared 'e'
               can be called directly for unknown ellipsoids.
        OC_w211_wome (wome) must be called before OC_w211_morth(). Set
               utn 'some' (must be non-zero). Segative means southern
hemisphere. Used to set the longitude of the central
               meridian (only used for utm to let/lon conversions)
        CC_w2ll_sorth (sorth) set the wim morth. Must be called before
               C_1211.
       CC_u2ll (east, lst, los) computes lst, los from east after CC_u2ll_morth() has already been called with morth.
    3. miscellaneous user interface service routines
       logitude of COORDINATE variable to seconds
                    decine 1
       latitude of COOKDIBLE variable to seconds
                    decimal
                - Convert an ALTEREC VARIABLE's spec to units and altitude
       convected - Convert the spherical coordinate character string of
                    a COORDINATE variable to decimal
       COORDINATE decimal lat/lon to stms
 ************************
finalude oprocess.b>
                                       /* Header for calls to MM-DOS
@include <stdio.b>
#include <ctype.h>
#define SQLCA_STORAGE_CLASS extern
                                      /* Switch for header files
                                                                          */
MINE SOL REGIN DECLARE SECTION:
                                       /* All SQL declarations are in
EXEC SQL INCLUDE hostwars.h;
                                       /* these header files
MIEC SQL INCLUDE faharris.h;
BORN SQL BED DECLARS SECTION;
EDEC SQL DECLUDE SQLCA;
#inglude "mean.h"
                                       /* Standard ASAN Sender File
#define abs (x) ((x)<0?-(x):(x))
#define RADIAMS_TO_SECONDS 206264.8062470964
#define #ECONDS_TO_RADIAMS 4.848136811095360-6
double sqrt(), sin(), cos();
static double al,a2,a3,a4;
                                /* lat coef: geodetic to rectifying */
static double a5 = 5.0e5;
                                 /* false easting (OTM) */
                                /* false morthing */
static double as:
static double a7 m 0.
                                 /* MOT */
static double as = .9996;
                                 /* UTM scale factor at central meridian */
static double a9;
                                 /* central meridian in seconds */
static double al0;
                                 /* radius of curvature */
static double all,al2,al2,al4; /* lat coef: rectifying to geodetic */
static double als:
                                 /* major axis */
static double alf:
                                /* eccentricity squared */
static double b1,b2,b3,b4;
                                /* intermediate values */
static double b5,b6,b7,b8;
static double b9,b10,b11,b12;
static struct ( cher *amo;
                 double a:
                                      /* semi-major axis */
                 double e;
                                      /* eccentricity squared */
                 } spheroid[] =
                     ( "australian",
                                         6378160.0,
                                                     0.0066945419,
                        "bessel",
                                         6377739.155, 0.0066743722,
                        "clark66".
                                         6378206.4, 0.006768658,
                        "clark#0".
                                         6378249.145, 0.0068035113,
                        "everest",
                                         6377276.345, 0.0066378466,
                        "international", 6378388.0, 0.00672267, "wgs72", 6378135.0, 0.006694317778
```

```
OC_get_spheroid -- Returns axis and excentricity squared
                     for the named spheroid. Return value
                     1 for success, 0 for failure to find
                     the spheroid in the list
 double *a, *e;
 ist m;
a = sizeof(spheroid)/sizeof(spheroid[0]);
#1.fdef CHRCHOUT
printf("\nget_spheroid to from td entries ", name, n);
while (n--)
  if (equal (name, spheroid[a].name)) {
    *a = spheroid[n].a;
    *e = spheroid[n].e;
    return 1:
    #1fdef CERCEOUT
      printf(" found 4d", a);
  #1fdef CEECEOUT
  printf(" mot found");
@endif
  return 0 ;
cher * CC_spheroid_name (n)
C_spheroid_name -- Find the name of spheriod n in the list
if (a < 0 || a >= sizeof(spheroid)/sizeof(spheroid[0])) return 0;
else return spheroid[n].neme;
static equal (a, b)
comparison between two character strings
                (Forced to lowercase)
******************
Char *a, *b;
char losse();
while (*a) if (losse (*a++) != losse (*b++)) return 0;
return *b == 0;
}
static char losse (c)
         -- Force character to lowercase
```

```
char c;
 if (a >= 'A' 66 a <= 'B') a += 'a' - 'A';
 int CC_u2ll_spheroid (spheroid_name)
     CC_w211_spheroid -- Set up spheroid persenters for UTM to Let/Long
                       given a spheroid name
 cher *spheroid_neme ;
 double a, a;
if (CC_get_spheroid (spheroid_neme, &e, &e))
return CC_u2ll_spheroid_peremeters (a, e);
                                            /* Enoug Spheroid */
 return -1;
                                             /* Vakaowa Spheroid */
int CC_u2ll_spheroid_parameters (a, e)
OC_w211_spheroid_persmeters -- Store the conversion persmeters
                      gives axis and excentricity
double s,e;
double x, x2, x3, x4:
if (a < 0.0 || a < 0.0 || a > 1.0) return -2;
                                            /* filegal values */
a16 - a:
= (((a * (7.0/32.0) + 5.0/16.0) * a + .5) * a + 1.0) * a * .25;
32 = 2 * 2;
크 * = * 크;
크 * = = * 크;
/* coefficients to convert geodetic to rectifying latitude */
al = -(((x * (195.0/64.0) + 3.25) * x + 3.75) * x + 3.0) * x = 3.0
a2 = (((1455.0/32.0) + x + 70.0/3.0) + x + 7.5) + x2;
a3 = -((70.0/3.0) + x + (945.0/0.0)) + x3;
= (315.0/4.0) \pm \pm 4;
/* coefficients to convert rectifying to geodetic latitude */
a11 = (((7.75 - (657.0/64.0) * z) * z - 5.25) * z + 3.0) * z;
a12 = ((5045.0/32.0) * x - (151.0/3.0)) * x + 10.5) * x2;
a13 = ((151.0/3.0) - (3291.0/8.0) * x) * x3;
a14 = (109.0/4.0) * x4;
/* radius of curvature */
a10 = (((225.0/64.0) * x2 + 2.25) * x2 + 1.0) * (1.0 - x2) * (1.0 - x) * a;
return 1:
int CC_u211_some (some)
OC_1211_1000
```

```
/* set false morthing (m6), compute central meridian (m3) */
15 (some < 0) {
   a6 = 10.006;
   whx = 30.0 + mose:
alsa {
   as = 0.0;
   wtz = 30.0 - mome;
m9 = (mts * 6.0 + 3.0) * 3600.0;
int OC w211 morth (morth)
CC_w211_morth
double morth;
double sizw, cosw;
double t, ts, ra, ra2, ra4, ra6, ra8;
b10 = ((aorth - a6) / a8 + a7) / a10;
if (abs(b10) > 1.47)
   return -1; /* rectifying let exceeds 1.47 radians, -84.15.30 */
sinw = sin(b10);
goew = gos (b10);
bl2 m goew * goew:
bl1 = (((a14 * b12 + a13) * b12 + a12) * b12 + a11) * sinw * cosw + b10;
siaw = sia (b11) ;
005W = 005 (b11);
ra = sqrt (1.0 - al6*sinw*sinw) * 1.0a6 / al5;
ra2 = ra * ra ;
m4 = m2 * m2 :
ra6 = ra2 * ra4 ;
   = 234 * 234 ;
    - sinw/cosw;
    = t + t:
b12 = coew * coew;
etas = a16 * b12 / (1.0 - a16);
b1
   = ZB/006W;
    ъ2
ьз
   ((c. + 24.0 + stas + 8.0 + 28.0) * ts + stas + 24.0 + stas + 6.0 + 5.0) * ts + sta4 / 120.0; = (((stas + 45.0 - 45.0) * ts + stas * 162.0 -90.0) * ts - stas * 107.0 - 61.0) * t * ra6 / 720.0;
ьв
b6
    = -(((ts * 720.0 + 1320.0) * ts
                    + 662.0) * to + 61.0) * b1 * rm6 / 5040.0 ;
  = (((ts + 1575.0 + 4095.0) + ts
                   + 3633.0) * ts + 1385.0) * t * xm8 / 40320.0;
return 1;
int OC u211 (east, lat. lon)
     00_1211
```

```
double east;
 double *let, *loa;
 double b9, b10;
b\theta = ((a5 - east) \pm 1.0e-6) / a8 ;
if (abe(b\theta) > a15 \pm 2.0e-7)
    return -1; /* utm easting to far from center of some */
b10 = b0 * b0 ;
 *let = (((b0 * b10 + b6) * b10 + b4) * b10 + b2) * b10 + b11)
                                                  * RADIANS TO SHOOMS ;
 *loa = (((b7 * b10 + b5) * b10 + b3) * b10 + b1) * b9
                                                  * RADIAMS_TO_SECONDS + m# ;
return 1:
 ist CC_ll2u (lst, los, east, morth, some)
 ∞_113±
double lat, lon;
double *east, *morth;
 int *some;
 int deg;
 double simp, cosp;
double stas, t, ts, ra;
double s1,s2,s3;
/* if some is i=0, force into this some, otherwise compute the some */
if (*some == 0) {
  if (lom < 0) {
    deg = -lom / 3600;
                              /* easters somes */
       *Econ = 31 + dag / 6;
   alse {
                              /* western somes */
      deg = los / 3600 ;
       *xone = 30 - deg / 6;
   if (lat < 0) *some = -(*some) ;
         /* Bow, set a6,29 */
OC_u211_some (*some);
if (abs(lat) > 302400.0) return -1; /* latitude above 84 degrees */
b10 = (a9 - loa) * SECONDS_TO_RADIAMS;

if (abs(b10) > .16) return -2; /* loagitude to far from center of whn some */
     = let * SECONDS_TO_RADIAMS;
; (6d) ate = qate
     = cos (b9);
= cos (b9);
= a15 / sqrt (1.0 - a16 * simp * simp);
= simp / cosp;
= t * t;
=
      = 00ep * 00ep;
œ
      = a1 * a1;
al
      = a1 * a2;
etas = a16 * a1 / (1.0 - a16);
ь1
      = IR * GOSP;
      = (1.0 - ts + etas) * b1 * c1 / 6.0;
= ((ts - 18.0) * ts + 5.0 + (14.0 - 58.0 * ts) * etas) * b1 * c2 / 120.0;
b5
      = (((179.0 -ts) * ts - 479.0) * ts + 61.0) * b1 * a3 / 5040.0;
ъ7
b12 = b10 * b10;
*east = (((b7 * b12 + b5) * b12 + b3) * b12 + b1) * b10 * a8 + a5;
```

```
= ma * al * t / 2.0:
       = (etas + (9.0 + 4.0 + etas) + 5.0 - ts) + b2 + c1 / 12.0;
       = ((ts - 58.0) * ts + 61.0 +
               (270.0 - 333.0 * te) * etas) * b2 * c2 / 360.0;
       = (((543.0 - ts) * ts - 3111.0) * ts + 1385.0) * b2 * a3 / 20160.0;
 *morth = (((b8 * b12 + b6) * b12 + b4) * b12 + b2) * b12 +
            ((((a4 * c1 + a3) * c1 + a2) * c1 + a1) * simp * coep + b9)
*morth = (*morth - a7) * a8 + a6;
return 1;
lon2dec -- Convert a COCKDINATE VARIABLE's longitude to
                       decimal (seconds of arc)
       Nortine unpacks longitude of the structure, reformsts it and them
       stores the reformatted and the converted value in the structure
00000180ATE *x;
double convocord():
register int 1;
Bifdef CENCROTT
printf("\mlom2dec ");
#endif
for (i = 0; i <14; i++) x->lon[i] = toupper(x->lon[i]);
if ((x->longitude = convocord(x->lon)) < OFF_EARTE) {
   if (x->longitude > 0.0) x->lon[12] = 'W';
else x->lon[12] = 'E';
else z->eastings = -1.0e75;
lat2deg(x)
/*********************************
                  -- Convert a COORDINATE VARIABLE's latitude to
                      decimal (seconds of arc)
       Routine unpacks latitude of the structure, reformats it and then
       stores the reformatted and the converted value in the structure
COORDINATE *x;
double comvecord();
register int 1;
for (i = 0; i < 14; i++) z->lat[i] = toupper(z->lat[i]);
z->lstitude = convocord(x->lst);
if (x->latitude < OFF_EARTE) (
  z->let[0] = ' ';
  if (x->latitude > 0.0) {
     1f (x->latitude > 324000.0) {
       #LOUTER (" > 90 ");
        2->morthings = -1.0e75;
     else x->lst[12] = 'W';
  alse {
     if (x->latitude < -324000.0) {
       MOUTES (" > 90 ");
z->northings = -1.0e75;
     alsa x->lat[12] = 'S';
```

```
else x->morthings = -1.0e75;
int altidec(x)
-- Convert an ALTSPEC VARIABLE's spen to units and
                     long integer value
      Note: Use only to convert values at the time of entry or the status
             line message will make you wonder with great smaxement .....
ALTEREC *x;
register int i, j;
for (i = 0; i < 10; i++) z->spec[i] = toupper(x->spec[i]);
j = strlem(x->spec);
x->altitude = 0;
for (1 = 0; 1 < j; 1++) {
   if (isspace(x->spac[i])) continue;
   if (isdigit(x->spec[i])) break;
   if (x=>xpec(x) == 'A') {
    stropy(x=>xpec, "AF ARROYD");
    stropy(x=>xeits, "TED");
else SLOUTES("Not altitude");
   return 0;}
12 (1 - j) (
   SLOUTER ("No value");
   return 0;}
   r>altitude = (10 * r>altitude) + r>spec[i] - 48;
i++; while (isdigit(r>spec[i]) && i < j);</pre>
for (; 1<j; 1++)
   if (lisspace(x->spac[i])) break;
TE (T -- 3) (
   SLOUTER ("Heed Reference (MSL, AGL, STC) or ASSIGNED");
   return 0:1
if (x->spec[i] == 'A') stropy(x->units, "AGL");
else if (x->spec[i] == 'M') stropy(x->units, "MSL");
else if (x->spec[i] == 'S') stropy(x->units, "SFC");
alsa return 0:
sprintf(z->spec, "404ld 4s", z->eltitude, z->units);
retura 0:
double conveced(x)
                     -- Unpack a character string with a spherical
                         coordinate and return seconds of arc as a double
      Note: The character string will be reformetted to nnnmm.'nn.m"X
            (me in 112x30'24.5"W)
cher z[];
int 1, j, ideg = 0, imin = 0;
double t = 0.0;
j = strlen(z);
for (1 = 0; 1 < j; 1++) (
                                              /* Degrees */
  if (isspace(x[i])) continue;
   if (isdigit(x[i])) break;
   SLOUTER ("Not a goordinate");
   return (double) OFF_RARTE; }
1f (1 - j) (
  SLOUTER ("No value");
  return (double) OFF_EARTE; }
```

```
ideg = (10 * ideg) + x[1] - 40;
     i++;} while (isdigit(x[i]) && i < j);
 if ((ideg > 180) {]  ((x[i] \ != ' \ ') \ 66 \ (x[i] \ != 'D') \ 66 \ (x[i] \ != ' \ ')') \ 66 \ (x[i] \ != -8))) \ \{
     SLOUTER ("? Degrees"); return (double) OFF HARTE;}
 12 (1 < 1-1) (
     144:
     for (; i < j; i++) {
   if (isspace(x[i])) continue;</pre>
                                                       /* Minutes */
        if (isdigit(x[i])) breek;
         SLOTTED ("? Minutes");
        return (double) OFT_EARTE;}
     for ( ;iedigit(x[i]) && i < j; i++) {
  imin = (10 * imin) + x[i] - 48;
    12 ((inin > 60) ||
((x[i] |= ' ') 46 (x[i] |= 'M') 46 (x[i] |= '\'') 46 (x[i] |= '\0'))) {
        SLOUTER ("? Minutes"); return (double) OFF_EARTE;)
 12 (1 < 1-1) (
     144;
     for (; 1 < j; 1++) {
                                                       /* Seconds */
        if (isspace(x[i])) continue;
        if (isdigit(x[i])) break;
        SLOUTEP ("? Seconds");
        return (double) OFF EARTE; }
     for (;isdigit(x[i]) && i < j;i++) {
        t = 10.0 * t + (double)(x[1]-48);
    if (x[i] = '.') (
     · 42 (4 < 3) {
           if (isdigit(x[i])) t += ((double)(x[i]-48))/10.0;
           1++:1
    if ((t > 60.0) || ((x[i] != '\0') 66 (x[i] != '') 66 (x[i] != '8') 66 (x[i] != '"'))) {
        SLOTTEP ("? Seconds"); return (double) OFF_EARTE;}
15 (1 < 1) 1++;
for (; 1 < j; 1++) (
                                               /* Hemisphere */
    if (lisspace(x[i])) break;}
if ((i = j) || (x[i] = 'H') || (x[i] = 'W')) {
    sprintf(x, *403d *02d' *04.11f\" ", ideg, imin, t);
    return (3600.0 * (double) (ideg) + 60.0 * (double) (imin) + t);}
else if ((x[i] = 'B') || (x[i] = 'H')) {
    sprintf(x, *403d *02d' *04.11f\" ", ideg, imin, t);
    return (-(3600.0 * (double) (ideg) + 60.0 * (double) (imin) + t));}
SLOUTEF ("? Hemisphere");
return (double) OFF_EARTE;
ist 000324tm(z)
COOR2utm -- Convert a decimal latitude and longitude to the
                        corresponding wim morthing/eastings for an ASAN
                         COORDINATE structure
COORDINATE *x;
char *clock();
if ( (x-)latitude < OFF_EARTE) 66 (x-)longitude < OFF_EARTE) ) ( if (CC_ll2u (x-)latitude, x-)longitude,
                 &z->eastings, &z->northings, &utmeone) == 1) return (int) 0;
```

```
/* If it didn't work, then complain to the user */
   else (
     SLOUTED ("Coordinate conversion error");
      fprintf(dante, "\nts COORDINATE Conversion error", slock());
      fprintf(dante, "\a\t\t\tLet = $12.21f [%s]", x->letitude, x->let);
      fprintf(dante, "\a\t\t\tLong = %12.21f [%s]", x->longitude, x->lon);
alsa (
  SLOTTEP ("Fix bad coordinates first");
return (int) -1;
/*
/*
          assa.pc -- ASAF Main Program
/*
/*
            1. Opens the printer.
            2. Calls strtASAW to establish communication with ORACLE
            3. Verifies that ASAN software is valid.
             4. Calls Uinit to start the screen driver.
             5. At end of the session closes detabase and printer
/
finalude <stdio.b>
                                /* The usual stuff, of course */
finalude oprocess.h>
                               /* Header for calls to MS-DOS */
#include <string.h>
                                /* String manipulation header */
Singlade <time.b>
MING SQL BRIGHT DUCLARE SUCTION;
EXEC SQL INCLOSE citvars.h;
EXEC SOL END DECLARS SECTION:
EXEC SQL INCLUDE SQLCX;
                                 /* SQL Communication Area
                                 /* Standard ASAN Beader file */
disclude "sees.b"
                                 /* In asan.pd it must be here */
                                 /* since we need to initialize */
                                    character arrays!
mein ()
extern int HEA DEBOG FEATURES;
int 1;
static cher *legal_motice[11] =
{"\a\a\a\a\t\t\t RESTRICTED RIGHTS LEGISD\a\a",
 '\tUse, duplication, or disclosure is subject to restrictions\n',
        as set forth in subdivision (b) (3) (ii) of the \n",
"\t
"\t Rights in Technical Data and Computer Software Clause\n",
"\£\£
       at 52,227-7013 of the DOD FAR Supplement.\a\a",
"\t\t\tBBM LABORATORIES INCORPORATED\2",
"\t\t\t 10 MOULTON STREET\n",
"\t\t\t Cambridge, MA 02238\n",
     (* 617-873-3000\n\n\n\n\n.")
User Interface Copyright (C) 1985, EMM Leboratories Incorporated\n",
*\t\t\t
         All Rights Reserved"};
FILE *fopen();
pra = fopea("pra", "a");
if (pra - NULL) (
   printf("\nCan't open printer!"); exit(128); }
dante = fopen("chronfil.sef", "s");
if (dente == NULL) {
   printf("\nCan't open chronfile("); exit(128); }
printf("40[2J", 27);
for (i = 0; i < 11; i++) printf("*s",legal_notice[i]);</pre>
printf("\a\a\a\a\a\a\a\t Please tap the space bar to continue, \"CTRL-C\"\
 to abort. \n\n");
do {
   i = getak();
   } while (i le ' ');
```

```
printf("ASAM Citation module starting....");
 MAN DEROG FEATURES = 1;
 stropy(cituid.arr, "INQUIRER/ALIGEIERI");
 cituid.lea = strlea(cituid.arr);
MING SQL COMMICT : cituid;
if (sqlom.sqloode) (
   printf("\mCannot Start Citation Detabase Enquiry due to\m\m*\m*\m*\m*\m*
           sqlos.sqlerm.sqlermo);
    exit (16);}
MING SQL SELECT COURT (entry num)
                FROM beadquerters .animal_list
                INTO :merenimals;
Tinit();
EXEC SQL BOLLBACK WORK RELEASE;
exit(0);
 query1.pd -- Routines for ditation database retrieval without keywords *
      This file contains:
      queryh - Set up for a new search
queryh - Select citations from the Ruman area
querys - Select citations from the Animal area
querys - Select citations from the Structures area
      deerla
deerle
      deerla
deerla
              - Select citations from the Modeling area
              - Select based on (first few letters of) Writer's name
      queryd
               - Select based on the range of Dates given
               - Select based on the occurrence of a phrase in the Title
/* Header for calls to MS-DOS */
dinglude 
⟨process.k>
finalude <stdio.b>
#define SQLCA_STORAGE_CLASS extern /* Switch for header files
EXCEC SQL REGIST DECLARS SECTION;
EXEC SOL INCLUDE sitvare.h:
MINEC SQL MED DECLARE SECTION;
MINE SOL INCLUDE SOLCA;
                                      /* Standard ASAN Reeder File
finclude "agen.h"
EXEC SQL DECLARE C100 CURSOR FOR
                      SELECT entry_num
                        FROM headquarters ditation_search
                       WEERE human_area = '2';
EXEC SQL DECLARS C101 CURSOR FOR
                      SELECT q.estry_num
                        FROM headquarters.citation_search s, qual_cit1 q
                       WHERE s.entry num = q.entry num
AND s.human_area = 'T';
EXEC SQL DECLARE C102 CURSOR FOR
                      SELECT q.entry_num
FROM beadquarters.ditation_search s, qual_dit2 q
                       MEDI s.entry num = q.entry num
AND s.human_area = 'T';
EXEC SQL DECLARE 0200 CURSOR FOR
                     SELECT entry_num
                       FROM headquarters.citation_search
                       WHERE enimi_eres = 'T';
EXEC SQL DECLARS C201 CURSOR FOR
```

and the second of the second o

188

SELECT q.entry\_aum FROM headquarters.citation search s, qual citi q REDE s.estry aux = q.estry aux AND s.esiml area = '7'; EXEC SOL DECLARE C202 CURSOR FOR SELECT q.estry ave. FROM headquarters.citation\_search s, qual\_cit2 q NEEDE s.entry num = q.entry num AND s.eniml\_erec = '7'; EXEC SQL DECLARE CHOO CURSOR FOR SELECT entry\_sum FROM headquarters.ditation\_search WEEKE struc\_area = '2'; MING SQL DECLARE CS01 CORSOR FOR SELECT q.estry\_num FROM headquarters.citation\_search s, qual\_cit1 q NEERE s.entry num = q.entry num AMD s.struc\_eren = 'T'; EXEC SQL DECLARE CHO2 CURSOR FOR SELECT q.entry\_num FROM headquarters ditation search s, qual\_dit2 q WHERE s.entry num = q.entry num AMD s.strnd area = 'T'; EXEC SOL DECLARE C400 CURSOR FOR SELECT entry\_ava FROM headquarters.citation\_search WEERE model\_area = '7'; MINC SQL DECLARE C401 CURSOR FOR FROM headquarters.citation\_search s, qual\_citl q
WERE s.entry\_num = q.entry\_num
AND s.model\_area = 'T'; MING SQL DECLARS C402 CURSOR FOR SELECT q.estry\_sum FROM beedquarters ditation\_search s, qual\_dit2 q MERE s.entry num = q.entry num AND s.model\_area = 'T'; EXEC SQL DECLARE CSO# CORSOR FOR SELECT authoraum FROM headquarters.author\_list a NEERE UPPER (author) LIKE :pettern; EXEC SQL DECLARE CSOO CURSOR FOR SELECT entry\_num FROM headquarters author ditation link WHERE authornum = :authornum; MANGE SOL DECLARS 0501 CURSOR FOR SELECT q.estry\_num FROM headquarters author citation link 1, qual cit1 q WEERE 1.estry num = q.entry num AND 1.authornum = :suthornum; MING SQL DECLARE 0502 CURSOR FOR

EXEC SQL DECLARE C600 CURSOR FOR

SELECT entry num

SELECT q.estry\_sum

FROM headquarters.citation\_search

WEERE 1.entry num = q.entry num AND 1.enthornum = :enthornum;

FROM headquarters suthor citation link 1, qual cit2 q

```
MEERE data_pub >= :data1
AND data_pub <= :data2;
 MING SGL DECLARE C601 CURSOR FOR
                        SELECT q.estry_sen
                         FROM beedquarters ditation search s, qual diti q
                        WHERE s.entry num = q.entry num
AND s.date_pub >= :date1
                          AND s.date_pub <= :date2;
EXEC SQL DECLARS 0602 CURSOR FOR
                        SELECT q.estry_num
                         FROM beadquarters.citation_search s, qual_cit2 q
                        WHERE s.entry_num = q.entry_num
AND s.date_pub >= :date1
                          AND s.date_pub <= :date2;
EXEC SGL DECLARS C700 CURSOR FOR
                       SELECT entry_num
                         FROM headquarters ditation titles
                        WHERE UPPER (title) LIKE :pettern;
MING SQL DECLARE C701 CURSOR FOR
                       SELECT q.entry_aum
                        FROM headquarters ditation titles d, qual ditl q
                        WEERE c.entry_num = q.entry_num
AND UFFER(c.title) LIKE :pettern;
MING SQL DECLARS C702 CURSOR FOR
                     SELECT q.entry_num
                        FROM headquarters citation titles c, qual_cit2 q
                        MEERS c.entry num = q.entry num
AND UFFER(c.title) LIEE :pettern;
ist quetup()
            ..
...............
           questup -- start-up routine that clears all temporary tables
                      for intermediate pointers in queries
#1fdef CHRCHOOT
   printf("\mqsetup ");
Seedil.
SLOUT("Furging obsolete qualifier lists");
EXEC SQL DELETE FROM QUAL_CIT1;
if (sqloa.sqloods)
   if (sqlca.sqlcode |= SQL_BOT) {
      BEDOMEG:
      SLOUTP ("ts", sqlos.sqlerms.sqlerms);
      aust (16);}
#1fdef CHRCHOUT
  else printf("\ngualifying table1 cleared %ld", sqlca.sqlcode);
Bend17
MINE SOL DELETE FROM QUAL CITY;
if (sqlos.sqloods)
   if (sqloa.sqloods != SQL_BOF) {
      ENDOMS:
      SLOUTS ("4s", sqloa.sqlorma.sqlormac);
      exit (16);}
#ifdef CEECEOUT
  else printf("\n@malifying table2 cleared %id", sqlca.sqlcode);
Seedif.
EXEC SQL COMMIT WORK:
qual entries = 0;
                                       /* No qualifying entries */
                                       /* No temporary table */
/* No date */
temptabl = 0:
dates1 = dates2 = 0;
selcrit[0] = '\0';
                                       /* No search path
animals = maranimals;
                                       /* ALL Animals ...
olddepth = 0;
contamm[0] = '\0';
```

```
majorst[0] = '\0';
minorat[0] = '\0';
address[0] = '\0';
setaff(6);
MENVALS ();
return (int) 0;
query's -- Create the subset of citations for the titles that
               pertain to the EURAN area
1mt 1:
#1fdef CENCHOUT
  printf("\mqueryh ");
-
SLOUT("Ermen Area Search");
stroat (selerit, "Reman ");
switch (temptabl) {
  0850 0:
     EXEC SQL OPEN C100;
     if (sqlos.sqloods) {
      . SLOUTS ("CURSOR G100");
        SLOUTS ("%s", sqlom.sqlorms.sqlorms);
     qual_matrice = 0;
     for (;;) (
        EXEC SQL FERCE C100 into : cause
       if (sqlos.sqloods == SQL_BOF) break; if (sqlos.sqloods) {
          EMDONES;
          SLOUTS ("CURSOR C100");
          SLOUTS ("te", sqlca.sqlarm.sqlarmc);
       EXEC SQL IMSERT INTO QUAL CIT1 (entry avm) VALUES (:enumb);
       if (sqlos.sqloods) (
          SLOUTS ("te", sqlos.sqlerm.sqlerme);
       #1fdef CESCEDUZ
          printf("\mAccepted ");
          for (i = 0; i < 5; i++) printf("4c", enumb.arr[i]);
       fendic
       qual_entries++;
       HENVALS ();
    KING SQL COMMET WORK;
    EXEC SQL CLOSE C100;
    temptabl = 1;
 case 1:
    EXEC SQL DELETE FROM QUAL_CIT2;
    EXCEC SQL COMMET WORK;
    EXEC SQL OPEN C101;
    if (sqloa.sqloods) (
       MINIOCHES :
       SLOUTS ("CURSOR C101");
       SLOUTS ("4s", sqlca.sqlerma.sqlerma);
    qual_entries = 0;
    for (;;) {
       MANGE SQL FETCH C101 into :extenb;
       if (sqlom.sqloods == SQL BOF) break; if (sqlom.sqloods) (
         MEDONAS ;
          SLOTTE ("CURSOR C101");
         SLOUTP ("4s", sqlos.sqlerms.sqlerms);
         exit (16) ;
      EXEC SQL INSERT INTO QUAL_CIT2 (entry_num) VALUES (:enumb);
```

```
if (sqlos.sqloods) {
MEDOMAG;
             SLOOTS ("4s", sqloa.sqlarma.sqlarma);
             emit(16);
          #1fdef CENCROUT
             printf("\minospted ");
             for (i = 0; i < 5; i++) printf("%c",enumb.arr[i]);
          foodif
          qual_entries++;
NEWVALS();
       EXEC SQL COMMET WORK;
       EXCEC SQL CLOSE C101;
       temptabl = 2:
       break:
       EXEC SQL DELEYS FROM GOAL CIVI;
       MING SQL COMMIT WORK;
       EXEC SQL OFFER C102;
       if (sqlos.sqloods) {
          BEDOMG;
          SLOUTS ("CURSOR C102");
          SLOUTP ("4s", sqlos.sqlerm.sqlermc);
          exit (16);
       qual_estries = 0;
       for (;;) {
          EXCEC SQL FETCE C102 into :enumb;
         if (sqloa.sqloods == SQL_BOF) break; if (sqloa.sqloods) {
             ESEDOMES;
            SLOUTS ("CURSOR C102");
            SLOUTS ("4s", sqlos.sqlorum.sqlorumc);
            erit (16) ;
         EXEC SQL DESERT DETO QUAL_CIT1 (entry_num) VALUES (:enumb);
         if (sqlos.sqloods) {
            SLOUTS ("%s", sqlos.sqlerm.sqlermso);
            emit (16);
         #1fdef CEBCEDOT
            printf("\maccepted ");
            for (1 = 0; 1 < 5; i++) printf("to", enumb.arr[i]);
         feedif
         HEWVALS ();
         qual_estries++;
      EXEC SQL COMMET WORK;
      EXEC SQL CLOSE C102;
   default:
     bed_temp();
#1fdef CEECEOUT
   printf(" *d estrics in table *d", qual_estrics, temptabl);
   #LOUTP ("Ack");
Feedif
return (int) qual_entries;
querys -- Create the subset of citations for the titles that
            pertain to the ANDRAL area
******************************
#1fdef CHECKOUT
  printf("\nquerya ");
#endif
```

```
SLOUT("Animal Area Secreb");
street (selerit, "Animal ");
switch (temptabl) {
   case 0:
     EXEC SQL OF ME C200;
      if (sqlom.sqloods) (
sLOUTS ("CURSOR C200");
         MEDCANSO;
         SLOUZE ("%s", sqloa.sqlarzm.sqlarzma);
      qual_estries = 0;
      for (;;) {
         MARC SQL PETCE C200 into :enumb;
         if (sqlox.sqloods == SQL_BOF) break; if (sqlox.sqloods) {
            SLOUTS ("CURSOR C200");
            ENDOMES:
            SLOTTP ("4s", sqlom.sqlarum.sqlarumo);
         EXEC SQL DESERT DETO QUAL_CIT1 (entry_num) VALUES (:enumb);
         if (sqlos.sqloods) {
            SLOUTP ("4s", sqlos.sqlerms.sqlerms);
         qual_entries++;
     EXEC SQL COLACT WORK;
      MING SQL CLOSE C200;
      temptabl = 1;
     break;
   case 1:
     MEET SQL DELETE FROM QUAL_CITE;
      EXEC SQL COMMET WORK;
      2000C SQL 0978F C201;
     if (sqlom.sqloods) {
        SLOUTP ("COLSOR C201");
         ENDOMEG:
         SLOUTS ("4s", sqloa.sqlorm.sqlormc);
     qual_estries = 0;
      for (;;) {
        EDEC SQL PETCE C201 into :enumb;
         if (sqlca.sqlcode - SQL_NOF) break;
         if (sqlos.sqloods) {
           BEDCHEG:
            SLOUTS ("CURSOR C201");
            SLOTT ("4s", sqloa.sqlarma.sqlarma);
        EXEC SQL DESERT DETO QUAL_CIT2 (entry_aum) VALUES (:enumb);
        if (sqloa.sqloods) {
            ENDOMEG;
            SLOTT ("4s", sqlos.sqlorm.sqlormo);
        qual_entries++;
     EXEC SQL COMMETT WORK;
     EXEC SQL CLOSE C201;
     temptabl = 2;
     break:
  Gese 2:
     EXEC SQL DELETE FROM QUAL_CIT1;
     EXEC SQL COMMET WORK;
     EXCEC SQL OFFER C202;
     if (sqlos.sqloods) {
        SLOUIS ("CURSOR C202");
        HINDOMS;
        SLOUTS ("4s", sqlca.sqlerma.sqlerma);
     qual_entries = 0;
     for (;;) {
        MICEC SQL PETCE C202 into :ente
        if (sqlom.sqloods == SQL_BOF) break; if (sqlom.sqloods) {
           SLOUTS ("CURSOR C202");
           ENDOMES;
           SLOUTS ("%s", sqloa.sqlarma.sqlarma);
        EXEC SQL INSERT INTO QUAL_GIT1 (entry_num) VALUES (:enumb);
        if (sqlca.sqlcode) {
           HINDOMSG:
```

```
SLOUTS ("4s", sqlos.sqlerra.sqlerras);
      EXEC SQL COMMET WORK:
      EXEC SOL CLOSE C202:
      temptabl = 1;
   default:
     bed_temp();
  printf(" %d entries in table %d", qual entries, temptabl);
-
return (int) qual_estries;
ist querys ()
querys -- Create the subset of citations for the titles that
               pertain to the STRUCTURES area
#1fdef CERCEOUT
  printf("\aquerys ");
SLOUT("Structures Area Search");
stroat (selemit, "Struct ");
switch (temptabl) {
  case 0:
     EXEC SQL 09EE 0300;
     if (sqloa.sqloods) (
        SLOUTS ("CURSOR C300");
        SLOUTP ("%s", sqlos.sqlerms.sqlermd);
     qual_estries = 0;
     for (;;) (
       EXEC SQL FETCE C300 into :enumb;
       if (sqlos.sqloods — SQL_EOF) breek; if (sqlos.sqloods) {
          ENDONSG;
          SLOUIS ("CURSOR C300");
          SLOUTS ("4s", sqlca.sqlarma.sqlarma);
       EXEC SQL DESERT DETO QUAL_CIT1 (entry_num) VALUES (:enumb);
       if (sqlca.sqlcode) {
          SLOUR ("%s", sqloa.sqlerm.sqlerme);
       quel_estries++;
    EXEC SQL COMMET WORK;
    EXCEC SQL CLOSE C300;
    temptabl = 1;
    break;
 case 1:
    MINEC SQL DELETE FROM QUAL_CIT2;
    EXEC SQL COMMET WORK;
    EXCEC SQL 09227 C301;
    if (sqlca.sqlcode) {
       SLOUTS ("CURSOR C301");
       ENDONESS:
       SLOUTS ("%s", sqlca.sqlerma.sqlermac);
    for (;;) (
      EXEC SQL FETCE C301 into :enumb;
      if (sqlos.sqloods = SQL EOF) break; if (sqlos.sqloods) {
         SLOUTS ("CORSOR C301");
         INDOMSG;
```

```
EXEC SQL INSERT INTO QUAL CIT2 (entry aum) VALUES (:enumb);
          if (sqloa.sqloods) {
             SLOUTE ("ts", sqlos.sqlerm.sqlerme);
          qual_estries++;
       MODE SQL COMMET WORK;
       MORE SQL CLOSE C301;
       temptabl = 2;
       breek;
    case 2:
      EXEC SQL DELETE FROM QUAL CIT1;
       EXEC SQL COMMET WORK;
       EXCEC SQL OPEN C302;
      if (sqlos.sqloods) {
stcors("cossos csoz");
         ENDOMS;
         SLOUTE ("4s", sqloa.sqlarma.sqlarma);
       for (;;) (
         EXEC SQL FETCH C302 into :esumb;
         if (eqlos.sqloods == SQL BOF) break;
if (sqlos.sqloods) {
    SLOUTS("CURSOR CHO2");
            MEDCHANG;
            SLOUTS ("4s", sqloa.sqlorms.sqlorms);
         EXEC SQL INSERT INTO QUAL_CIT1 (entry_num) VALUES (:enumb);
         SLOUIS ("ts", sqlos.sqlerm.sqlermed);
         qual_catrics++;
      MEMO SQL COMMENT WORK;
      MORC SQL CLOSE C302;
      temptabl = 1;
   default:
      bed_temp();
#1fdef CENCEOUT
   printf(" %d entries in table %d", qual_entries, temptabl);
Sendif.
return (int) qual_entries;
ist querys()
querym -- Create the subset of citations for the titles that
               pertain to the MCDELING area
*******************************
#1fdef CHBCHOUT
  pristf("\nquerym ");
SLOUT ("Model Area Search");
street (selerit, "Model ");
switch (temptabl) {
  0850 0:
     EXEC SQL OFFEE C400;
     if (sqloa.sqloods) {
        SLOURS ("CURSOR C400");
        EDIDOMSG;
        SLOUTS ("4s", sqlca.sqlerma.sqlerma);
```

SLOTTO ("4s", sqlos.sqlerms.sqlerms);

```
qual_estries = 0;
    for (;;) (
       EXEC SQL FEECE 0400 into :enumb;
       if (sqloa.sqloode == SQL_EOF) break; if (sqloa.sqloode) {
          SLOURS ("CURSOR C400");
          SLOUTP ("4s", sqloa.sqlorma.sqlormo);
       EXEC SQL DESERT DFTO QUAL_CIT1 (entry_num) VALUES (:enumb);
       if (sqlca.sqlcode) {
          SLOUP ("%s", sqlom.sqlomms);
       qual_estries++;
    EXEC SQL COMMETT WORK;
    EXEC SQL CLOSE C400;
    temptabl = 1;
    breek:
 case 1:
    MING SQL DELETE FROM QUAL_CITE;
    EXEC SQL COMMITT WORK;
    EXEC SQL OPEN C401;
    if (sqlam.sqloods) (
       SLOUTP ("CURSOR C401");
       SLOUTF ("to", sqlos.sqlerms.sqlerms);
    qual entries = 0;
    for (;;) (
       EXEC SQL FETCE C401 into :enumb;
       if (sqlom.sqloods = SQL_BOF) break;
if (sqlom.sqloods) (
         ELOUTE ("CURSOR C401");
          EMDCMSG;
          SLOUTS ("%s", sqlox.sqlorrm.sqlorrmc);
       EXEC SGL INSERT INTO GUAL_CITY (entry_aum) VALUES (:enumb);
       if (sqlos.sqloods) {
          EMPONES:
          SLOUTS ("4s", sqloa.sqlarma.sqlarma);
       qual_estrics++;
   MINC SQL COMMITT WORK;
   KNEC SOL CLOSE C401:
   temptabl = 2;
   breek;
case 2:
   EXEC SQL DELETE FROM QUAL_CIT1;
   EXEC SOL COMMITT WORK:
   EXEC SQL OFFER C402;
   if (sqlca.sqlcode) {
      SLOUTS ("CORSOR C402");
      ENDOMES:
      SLOUTS ("4s", sqloa.sqlorum.sqlorum.c);
   qual entries = 0;
   for (;;) (
      MIEC SQL FETCE C402 into :enumb;
      if (sqlca.sqlcode == SQL_BOF) break; if (sqlca.sqlcode) (
         SLOUTS ("CURSOR C402");
         MEDCHAS;
         SLOUIP ("4s", sqlca.sqlerm.sqlermc);
      EXEC SQL DESERT DITO QUAL_CIT1 (entry_num) VALUES (:enumb);
      if (sqlca.sqlcode) (
         ENDOMES:
         SLOUTS ("4s", sqlca.sqlerma.sqlerma);
      qual_entries++;
   MICHE SQL COMMET WORK;
   EXEC SQL CLOSE C402;
   temptabl = 1;
  break;
default:
```

```
bed_temp();
#1fdef CEBCROUT
  printf(" %d estrics in table %d", qual_estrics, temptabl);
Bendi f
return (int) qual_entries;
int queryw()
quaryw -- Create the subset of mitations for the author(s) whose
              name(s) look like the pettern given in "authornem"
register int i, j;
Bifdef CENCEDUT
   printf("\nqueryw ");
for (i = 0; i < authornom.len; i++) {
   if (authornem.arr[i] != ' ') break;
for (j = 1; j < authornem.lem; j++)
     authornem.arr[j-1] = authornem.arr[j];
   authornen.len--;
   authornes.arr[authornes.len] = '\0';
   #1fdef CEECEOUT
     printf("\n*s (*u)", authornem.arr, authornem.len);
      SLOUTS ("Ack");
   Seedif.
if (!authornam.lem) return (int) -1;
stropy(pettern.arr, authornem.arr);
street (pettern.arr, "9");
patters.lea = strlen(patters.arr);
for (i = 0; i < pattern.len; i++) pattern.arr[i] = toupper(pattern.arr[i]);
SLOUP("Author Search");
street (selerit, pettern.err);
MANUE SQL OFFER C50s;
if (sqlos.sqloods) (
   SLOUTS ("CURSOR C50s");
   SLOOTS ("to", sqloa.sqlarma.sqlarma);
   exit (16);
qual_entries = 0;
1f (temptabl - 1) (
  EXEC SQL DELEYS FROM QUAL_CITE;
alsa (
  MINE SOL DELETE FROM QUAL CITI;
MING SQL COMMIT WORK;
EXEC SQL FETCE C50s into :authornum;
if (sqlos.sqloods) (
if (sqlos.sqloods != SQL_BOF) (
EMBORREG;
      SLOUTS ("CURSOR CSOs");
     SLOUTS ("%s", sqloa.sqlerma.sqlermac);
     erit (16) ;
     }
olso {
     #1fdef CHECKOUT
        printf("\nComparing table %d for Author number ", temptabl);
        for (i = 0; i < 5; i++) printf("*d", authornum.arr[i]);</pre>
     Fendis.
     switch (temptabl) {
     case 0:
```

```
EXEC SOL OF ME C500;
    if (sqlqa.sqloode) {
       #LOUZE ("CURSOR C500");
       SLOURS ("%s", sqloa.sqlarm.sqlarmc);
       exit (16);
    for (;;) {
       EXEC SQL FETCE C500 into :enumb;
       if (sqlos.sqloods - SQL_MOF) break;
       if (sqloa.sqloods) {
          #1.0029 ("CURSOR 0500");
          ENDOMS;
          SLOUTP ("ts", sqlox.sqloxxx.sqloxxxc);
          emit (16);
       EXEC SQL DESERT DITO QUAL_CIT1 (entry_aum) VALUES (:enumb);
       if (sqlos.sqloods) {
          SLOTTP ("4s", sqlom.sqlarra.sqlarrac);
          exit (16);
       #1fdef CENCROTT
         printf("\mAccepted ");
          for (i = 0; i < 5; i++) printf("40",camb.arr[i]);
       Sendif.
       EXEC SQL COMMETT WORK;
       if (sqlca.sqlcode) {
         EDCOMG;
         SLOUTP ("%s", sqlox.sqlorms.sqlorms);
          azit (16) ;
      qual_entries++;
   EXEC SQL CLOSE C500;
   breek:
case 1:
   EXEC SQL OPEN C501;
   if (sqloa.sqloods) (
      SLOUTP ("CURSOR C501");
      SLOUTS ("ts", sqloa.sqlarma.sqlarma);
   for (;;) {
      EXEC SQL FETCE C501 into :enumb;
      if (sqlom.sqloods — SQL ROF) break;
if (sqlom.sqloods) {
         SLOUTS ("CORSOR C501");
         ENDOMES:
         SLOTTO ("4s", sqloa.sqlarma.sqlarma);
         axit (16) ;
      EXEC SQL DESERT DETO QUAL_CIT2 (entry_num) VALUES (:enumb);
      if (sqlos.sqloods) {
         ENDOMSG;
         SLOUTP ("4s", sqlox.sqloxma.sqloxma);
      #1fdef CERCEOUT
         printf("\nAccepted ");
         for (i = 0; i < 5; i++) printf("4c",eaumb.arr[i]);
      -
      NUMBER OF COMMENT WORK;
      if (sqloa.sqloods) {
         EMDOMAG;
         SLOOTS ("4s", sqloa.sqlarms.sqlarms);
         exit (16);
      qual estrice++;
      MENVALS () ;
   EXEC SQL CLOSE C501;
  breek:
Case 2:
  MINIC SQL OFFEE C502;
   if (sqlos.sqloods) {
     ENDOMSG;
      SLOTER ("CURSOR CS02");
      SLOUTS ("4s", sqloa.sqlerra.sqlerrad);
      emit (16) ;
```

```
}
for (;;) {
            EXEC SQL FETCE C502 into :enumb;
            if (sqloa.sqloods == SQL_BOF) break;
if (sqloa.sqloods) {
               SLOUZP ("CURSOR CS02");
               EDOMG:
               SLOUPP ("ts", sqloa.sqlerma.sqlerma);
               emit (16);
            MING SQL IMMENT INTO QUAL_CIT1 (entry_num) VALUES (:enumb);
            if (sqlos.sqloods) {
               SLOUTS ("4s", sqloa.sqlerm.sqlerme);
               exit (16);
            #1fdef CENCHOUT
              printf("\macompted ");
               for (i = 0; i < 5; i++) printf("%s",eaumb.arr[i]);
            EXEC SQL COMMIT WORK;
            if (sqlom.sqloods) {
              MEDCHAS;
              SLOUTS ("4s", sqlox.sqlerm.sqlerme);
              exit (16) :
            qual estrice++;
           MANUALE () ;
         MINC SQL CLOSE C502;
         break:
      default:
         bed_temp();
      EXEC SQL FETCE C50s INTO :authoraum;
      } while (!sqlca.sqlcode);
   if (sqloa.sqloode != SQL_BOF) (
      ENDOMSG;
      SLOTTE ("CURSOR CSOS");
      SLOUTS ("4s", sqlca.sqlerum.sqlerumc);
      exit (16);
EXEC SOL CLOSE CSOs:
#1fdef CHBCHOUT
   printf("\sold table %d", temptabl);
-
temptabl = (temptabl + 2) + 1;
Sifdet CHICKOOT
  printf(" %d entries in table %d", qual_entries, temptabl);
   d12
return (int) qual_entries;
queryd -- Create the subset of citations for citations that were
               published between two years
Bifdef CENCHOUT
  printf("\nqueryd ");
-
if (dates1 > dates2) {
                                   /* Check on the order of the dates */
  if (detec2) (
     sprintf(date1.arr, "44d", dates2);
                                              /* They are out of order */
     sprintf(date2.arr, "44d", dates1);}
  else (
     sprintf(date1.arr,"%4d",dates1);
stropy(date2.arr,"2000");}
                                             /* Omly one date entered */
alsa {
  if (detect) (
                                   /* There is a second date specified */
     sprintf(date1.arr, "44d", dates1);
                                                /* They are in order */
     sprintf(date2.arr, "44d", dates2);}
```

```
/* They are both sero! (or negative) */
   else return (int) -1;
date1.lem = date2.lem = 4;
street (selexit, "/");
streat (selerit, datel.arr);
street (selerit, "/");
streat (selerit, date2.err);
SLOUT("Date Search");
#1fdef CEBCEOUT
  printf(" Dates to and to", datel.arr, datel.arr);
switch (temptabl) {
   case 0:
      EXEC SQL OPEN C600;
      if (sqlos.sqloods) {
         SLOUTS ("CURSOR GEOO");
         SLOUTS ("ts", sqloa.sqlarm.sqlarmo);
      qual_estries = 0;
      for (;;) {
         EXEC SQL FETCE C600 into :enumb;
         if (sqlos.sqloods == SQL_BOF) break; if (sqlos.sqloods) {
            SLOUTS ("CORSOR C600");
            THE COMME
            SLOUTS ("4s", sqloa.sqlarma.sqlarmac);
         EXEC SQL DESERT DIFO QUAL_CIT1 (entry_aum) VALUES (:enumb);
         if (sqlca.sqlcode) {
            HENDOMSG;
            SLOUTP ("%s", sqloa.sqlerrm.sqlerrms);
         qual_estries++;
      MINDS SQL COMMENT WORK;
      MANUE SQL CLOSE C600;
     temptabl = 1;
break;
   case 1:
     MINEC SQL DELETE FROM QUAL_CIT2;
      EXEC SQL COMMETT WORK;
      MOUSE SQL OFFER C601;
      if (sqlam.sqloode) {
         SLOUTS ("CORSOR GS01");
         INDOMS;
         SLOUTS ("ts", sqloa.sqlerm.sqlermo);
      qual_entries = 0;
      for (;;) (
         EXEC SQL FETCE C601 into :enumb;
        if (sqlox.sqloods == SQL_EOF) break; if (sqlox.sqloods) {
           SLOTTE ("CORSOR C601");
            ERECOMES:
            SLOUTE ("4s", sqlos.sqlerm.sqlermo);
        EXEC SQL DESERT DETO QUAL_CIT2 (entry_num) VALUES (:enumb);
        if (sqlos.sqloods) {
EMDOMSS;
            SLOTT ("%s", sqlox.sqlerrm.sqlerrmc);
        qual_entries++;
     EXEC SOL COMMET WORK:
     EXEC SOL CLOSE C601:
     temptabl = 2;
  0850 2:
     EXEC SQL DELETE FROM QUAL_CIT1;
EXEC SQL COMMIT WORK;
     EXEC SQL OPEN C602;
     if (sqlca.sqlcode) {
        SLOUTS ("CURSOR C602");
        1000046G;
        SLOUTP ("4s", sqloa.sqlerma.sqlerma);
     qual_estrice = 0;
```

```
for (;;) {
           MANUE SQL PERCE C602 into :enu
           if (sqlos.sqloods - SQL_BOF) breek;
           if (sqlcs.sqlcode) {
    sLcore("corses c602");
              ENDOMAS;
              SLOUTS ("ts", sqloa.sqlerm.sqlerme);
           MING SQL INSERT INTO QUAL_CIT1 (entry_num) VALUES (:enumb);
           if (sqlos.sqloods) (
             HIDOMEG:
             SLOUTS ("%s", sqlos.sqlerms.sqlerms);
          qual_entries++;
       MING SQL COMMITT WORK;
       MINE SQL CLOSE C602;
       temptabl = 1;
       break:
    defealt:
       bed_temp();
 fifdef CENCENTY
   printf(" %d estries in table %d", qual_estries, temptabl);
 -
 return (int) qual_estrice;
 queryt -- Create the subset of citations for the titles that
                 tain the phrase given in "title_frag"
 ***********************************
 register int i, j;
 #ifdef CENCROOF
   printf("\aqueryt ");
 #endif
titlefreg.lem = strlem(titlefreg.mrr);
for (i = 0; i < titlefrag.len; i++) {
   if (titlefrag.arr[i] != ' ') break;
   for (j = 1; j < titlefreg.len; j++)
      titlefrag.arr[j-1] = titlefrag.arr[j];
   titlefreg.len--;
   titlefrag.arr[titlefrag.len] = '\0';
   #1fdef CENCEDOT
     printf("\nêu ês", titlefrag.len, titlefrag.arr);
   feadle
if (!titlefrag.lem) return (int) -1;
stropy(pettern.arr, "%");
stroat(pettern.arr, titlefreg.arr);
stroat (patters .arr, "4");
pettern.len = strlen(pettern.err);
for (i = 0; i < pettarn.len; i++) pettarn.arr[i] = toupper(pettarn.arr[i]);
stroat (selerit, pattern.arr);
#1fdef CERCHOUT
  printf("\%u %s", pattern.les, pattern.err);
SLOUT("Title Search");
switch (temptabl) {
  GR.50 0:
     EXCEC SQL 09301 0700;
      if (sqlos.sqloods) (
        SLOUZE ("CURSOR C700");
        SLOUIS ("4s", sqlos.sqlerrm.sqlerrmd);
     qual entries = 0;
     for (;;) {
EDEC SQL FERCE C700 into :enumb;
        if (sqlca.sqlcode == sqL_BOF) break;
if (sqlca.sqlcode) {
           #LOUSE ("CURSOR C700");
```

```
BEDOMES;
              SLOUTS ("%s", sqloa.sqlarms.sqlarms);
           RESIDE SQL DESERT DETO QUAL_CIT1 (entry_num) VALUES (:estumb);
           if (sqloa.sqloods) {
              SLOUTP ("ts", sqloa.sqlarm.sqlarmo);
           qual_estries++;
       EXCEC SQL COMMETT WORK;
       EXEC SQL CLOSE C700;
       temptabl = 1;
break;
    case 1:
       MORE MOL DELETE FROM QUAL_CIT2;
       EXEC SQL COLORET WORK;
       EXEC SQL OPEN C701;
       if (sqlos.sqloods) {
          SLOUTP ("CORSOR C701");
           SLOUTS ("%s", sqloa.sqlerms.sqlerms);
       qual_estries = 0;
       for (;;) (
          EXEC SQL FETCE C701 into :enumb;
          if (sqlom.sqloods == SQL_MOF) break;
if (sqlom.sqloods) {
             SLOUTS ("CURSOR C701");
             SECONDS:
             SLOUTS ("%s", sqloa.sqlerm.sqlerme);
          NORC SQL INSERT INTO QUAL_CIT2 (entry_num) VALUES (:enumb);
          if (sqlos.sqloods) {
              SLOUTS ("4s", sqlos.sqlerms.sqlerms);
          qual_estries++;
       EXEC SQL COMMET WORK;
       EXCEC SQL CLOSE C701;
       temptabl = 2;
break;
    cese 2:
      MESC SQL DELETE FROM QUAL_CIT1;
EXEC SQL COMMIT WORK;
       EXCEC SQL OFFEE C702;
       if (eqloa.sqloode) {
          ENDOMES:
          SLOUTS ("CORSOR C702");
          SLOUTS ("%s", sqloa.sqlerma.sqlerma);
       for (;;) (
         EXEC SQL FETCE C702 into :enumb;
         if (sqloa.sqloods == SQL_BOF) break;
if (sqloa.sqloods) {
    sLOOTS ("CURSOR C702");
             HINDOMS;
             SLOUTS ("4s", sqloa.sqlarma.sqlarma);
         EXEC SQL DESERT DETO QUAL_CIT1 (entry_num) VALUES (:enumb);
         if (sqlca.sqlcode) (
             SLOOM ("4s", sqlom.sqlarm.sqlarma);
         qual_estries++;
      MATEC SQL COMMETT WORK;
      EXEC SQL CLOSE C702;
      temptabl = 1;
   default:
     bed_temp();
#1fdef CENCEOUT
  printf(" %d entries in table %d", qual_entries, temptabl);
Bendit!
```

```
return (int) qual_entries;
 int bed_temp()
 thar mag[60];
 sprintf(msg, "Bed temporary table identifier 45d used!", temptabl);
 SLOUTED (mag) ;
 exit (16) :
 /****************************
       queryal.pd -- Noutines for ditation database retrieval in the
                    animal area using the taxonomy table
      This file contains:
#include  ca.b>
                                   /* Seeder for calls to MS-DOS
 #include <stdio.b>
#imalude <atype.h>
Midefine SQLCA STORAGE CLASS exters /* Switch for header files
MANG SQL BROTH DECLARS SECTION;
EXEC SQL RECEIPE ditvars.h;
VARCEAR foo[14];
                                    /* Dummy for uniqueness fetches */
EXEC SQL END DECLARE SECTION;
REEC SQL INCLUDE SQLCA;
#include "asen.h"
                                   /* Standard ASAN Seeder File
EXEC SOL DECLARE CA001 CURSOR FOR
                     SELECT anal_id, anal_ness
                      FROM headquarters animal list
                      WHERE amal name LIKE :animal;
EXEC SQL DECLARE CASO2 CURSOR FOR
                     SELECT annimans, annimal list
                      WHERE smal_id LIEE :petters
                      ORDER BY smallid, small seme;
EXEC SQL DECLARE CA100 CORSOR FOR
                      SELECT amal_id
                       FROM beadquarters . animal list
                      WHERE and id LIKE :pettern;
ROBC SQL DECLARE CAllO CURSOR FOR
                     SELECT e.entry_num
FROM beadquarters.enimal_effects e
                      WEERE and id - : nexternl;
MING SOL DECLARE CAll! GURSON FOR
                     SELECT e.entry_num
                       FROM bendquarters animal effects e, qual citl q
                      WHERE e.entry num = q.entry num
                        AND amalid = :nextern1;
EXEC SQL DECLARE CAll2 CURSOR FOR
                     SELECT o.estry_swn
                      FROM headquarters animal effects o, qual cit2 q
REFRE c.cetry num = q.cetry num
AND annl id = :nextern1;
int mknomlet()
/***********************************
```

```
minmlet -- Create the subset of animals that belong to the next
          level down in the texonomy table
cher *clock();
FILE *tatblkf, *fopen();
int accent, depth, i, lowdepth;
static char *dots[] = (" ", ". ", ".. ", "... ");
#1fdef CENCEOUT
  printf("\mmkammlst ");
Seedif.
if ( (txtblkf = fopen("txtblk\\vereainl.txt", "w")) == MULL) {
  SLOUZED("Error creating next level help window");

Sprintf(dante, "\n*s could not open animal textblock file", slock());
   return (int) DOFLICATE_OBJECT;
MINE SOL OFFER CAOCI:
                                         /* Find the "object" */
if (sqloa.sqloods) {
   HEROCHASC:
   SLOUTP ("CURSOR CA001") :
   SLOUIS ("4s", sqloa.sqlerma.sqlerma);
accust = 0;
lowdepth = 16;
for (;;) {
  EXEC SQL FETCE CA001 1870 :anml_id, :thisbeast;
  if (sqlom.sqloods — SQL_BOF) break; if (sqlom.sqloods) {
     ENDOMES:
     SLOUTS ("FETCE CA001");
     SLOUTS ("ts", sqlos.sqlorm.sqlorma);
  thisbeast.arr[thisbeast.len] = '\0';
  #1fdef CERCEDUT
     nml_id.arr[aml_id.lea] = '\0';
     printf("\minimal_id is %s", amml_id.arr, thisbeast.arr);
  Bead14
  fprintf(txtblkf, "Animals below %s are:\a\a", thisbeast.arr);
  sprintf(workspace.arr, "Looking for Animals below to ", thisbeast.arr);
  SLOUT (workspace.arr);
  for (1 = 6; 1 > 0;) {
                                         /* Where are we? */
     if ((amal_id.arr[i] != '0') || (amal_id.arr[i-1] != '0')) break;
     1 - 2:1
  depth = 1;
  if (depth < olddepth) continue;
  if (lowdepth > depth) lowdepth = depth;
  fifdet CENCROOT
    printf(" depth 4d", depth);
  Pendif
  if (depth < 6) {
                                        /* You can't go further than that */
     for (1 = 0; 1 <= depth; 1++)
       patterm.arr[i] = anml_id.arr[i];
     pattern.arr[i] = '%';
     pettern.len = i+1;
      pettern.arr[pettern.len] = '\0';
        printf(" Pattern is %s", pattern.arr);
     Seedif.
     MADEC SQL OF THE CA.002;
                                           /* What else is there like it */
     if (sqloa.sqloods) {
       ENDOMES;
        SLOUTS ("CURSOR CA002");
       SLOUTS ("4s", sqloa.sqlarma.sqlarma);
     for (;;) {
       EXEC SQL FETCE CA002 INTO :anninese, :nexterni;
       if (sqlca.sqlcode) break;
     #1fdef CRECEOUT
       nextanni.arr[nextanni.len] = '\0';
```

```
m.arr[anninema.len] = '\0';
         printf("\aFound to to", nextennl.arr, annhome.arr);
     Bendiff.
        if (!strnicep(nextern1.arr, smal_id.arr, 10)) continue;
        for (i = 6; i > 0;) {
if ((martannl.arr[i] != '0') || (martannl.arr[i-1] != '0')) break;
     Bifdef CENCEDUT
       printf(" included");
     Seedil!
        anminame.arr[anminame.len] = '\0';
fprintf(txtblkf, "4s 4s\a", dots[i/2], anminame.arr);
         scount++;
      12 (sqlos.sqloods != SQL_BOF) {
         Sprintf (deste,
                 "\mes &s\makes\t\t\t looking for animals like &s",
                 clock(), sqlcs.sqlcrms, sqlcrms, eximal.arr);
      fprintf(tatblkf, "\a\a"):
      EDEBC SQL CLOSE CA002;
if (sqlos.sqloods != SQL_BOF) (
   EMDOMES:
   foristf (deste.
           "\a4s 4s\a\t\t\t looking for enimals like 4s",
           clock(), sqlca.sqlerm.sqlermc, animal.arr);}
fprintf (txtblkf, "
                      - = -\a*);
falose(txtblkf);
sprintf(workspace.arr, "4d entries of 4d for 4s", account, animals, animal.arr);
fifdef CENCHOT
SLOUTS (workspace.arr);
#endi#
if ((lowdepth (= 16) && (lowdepth > olddepth)) olddepth = lowdepth;
if (olddepth < 8) olddepth += 2;
return (int) accust;
int queryull()
quaryall -- Find the animal of "species 1" in the taxonomy table
1mt 1:
  printf("\nqueryall ");
Bendif
                               /* Current Search depth is zero */
/* Current animal not yet defined */
olddepth = 0:
amal_id.arr[0] = '\0';
amal_14.1ea
species1.lea = strlea(species1.arr);
for (i = 0; i < species1.len; i++) {
   aminel.arr[i] = speciesi.arr[i];
   if (isspace(speciesl.arr[i])) break;
if (i == 0) return (int) 0;
                                     /* Animal not entered */
maimal.mrr(1) = '4'; 1++;
mainel.err[1] = '\0';
enimal.les = 1;
if (animals - mkarmlet()) (
                                                /* Find the second level */
  NEW SCREEN ("akeyalt");
   ADD_WINDOW("speciatry", 5, 1);
   return (int) 0; }
int queryal2()
```

```
queryul2 -- Find the animal of "species 2" in the taxonomy table
 int count. i:
 #1fdef CHICKNOT
   printf("\nquerym12 ");
   215
species2.lea = strlem(species2.arr);
for (i = 0; i < species2.lea; i++) {
    animal.arr[i] = species2.arr[i];</pre>
   if (isspace(species2.arr[i])) break;
12 (1 I= 0) (
                               /* Animal entered ? */
   mainel.arr[1] = '4'; 1++;
   mainel.arr[i] = '\0';
   emimal.len = 1;
   if (count = mkanmlet()) {
     manuals = count;
     REMOVE_WINDOW();
     NEW_SCREEN ("akeyalt");
     ADD_WINDOW("speciatry", 6, 1);
     return (int) 0; }
REMOVE WINDOW();
NEW_SCREEN ("altoysrch");
ist queryal3()
queryal3 -- Find the animal of "species 3" in the taxonomy table
int count, 1:
#ifdef CEECHOUT
  printf("\mquerym13 ");
Bendif
species3.len = strlen(species3.arr);
for (i = 0; i < species3.len; i++) {
    animal.arr[i] = species3.arr[i];</pre>
  if (isspace(species).arr[i])) break;
12 (1 I= 0) {
                          /* Animal entered ? */
  animal.arr[i] = '$'; i++;
animal.arr[i] = '\0';
  animal.len = 1;
  if (count = mkanmlet()) {
     animals = count;
    REMOVE WINDOW();
     MEM_SCREEN("akeyalt");
     ADD_WINDOW("speciatry", 7, 1);
     return (int) 0;}
  }
REMOVE WINDOW();
MEN_SCREEN ("akeyerch");
int queryal4()
queryul4 -- Find the animal of "species 4" in the taxonomy table
int count, 1;
```

```
#1fdef CHRCHOUT
   printf("\mquerym14 ");
  Seedif
 for (i = 0; i < species4.len; i++) {
    animal.arr[i] = species3.arr[i];</pre>
    if (isspace(species4.arr[i])) break;
 12 (1 I= 0) (
                               /* Jaimel entered ? */
    maimal.azz[i] = '4'; i++;
    minel.arr[1] = '\0';
    nainel.len = 1;
if (count = mknumlet()) {
       animals = count;
       }
 REMOVE_WINDOW();
 MEN SCHEM ("akeysrob");
 int asrah002()
 /************************************
       asrch002 - Search based on entries where an Animal is select
 Bifdef CHRCHOUT
 printf("\masrch002 ");
 |
|-mdif
 re (dmezia()) (
   HENVALS ();
   if (queryt()) (
      querya10();
      )
   }
  HVALS ();
 if ([qual_estries) {
    SLOTER("No estries metching these criteria were found");
   return (int) 0;}
else {
   if (|malisto()) {
      1f (|shwartait()) {
    maw_scream("citdspl");
         ADD_WINDOW("citdispaction", 19, 1);}
   return (int) 0;}
ist querye10()
      queryal0 - Select citations that deal with specific animal(s)
1st 1;
fifdef CEBCEOUT
  printf("\aquerya10 ");
Seedil.
if (olddepth == 0) return (int) qual_entries;
straopy(pattern.arr, ammi_id.arr, (unsigned int) (olddepth-2));
pattern.arr[olddepth-1] = '*';
pattern.arr[olddepth] = '\0';
pettern.len = strlen(pettern.arr);
street (selerit, petters.err);
#1fdef CEECEOUT
  amml_id.arr[arml_id.len] = '\0';
```

```
printf("\ats, use tu characters for ts", amal_id.arr, i, pettern.arr);
 -
SLOUT("Specific Animal Search");
EXCEC SQL OFFER CA100;
   if (sqlca.sqlcode) {
      SLOUTS ("CORSOR CA100");
      ENDOMES:
      SLOUTS ("4s", sqloa.sqlerma.sqlerma);
qual_entries = 0;
EXEC SQL PERCE CA100 into :nextern1;
if (sqlos.sqloods - sql_BOF) (
   MINE SOL CLOSE CA100;
   return (int) qual_estries;}
if (sqlox.sqloods) (
   SLOUTS ("FETCE CALOO");
   SLOTTS ("4s", sqlos.sqlerm.sqlerma);
  else if (temptabl == 2) (
     MESC SQL DELETE PROM GOAL CIT1; }
  EXEC SQL COMMIT WORK;
#1fdef CERCEOUT
  mexterml.arr[nexterml.len] = '\0';
  printf("\nTaxon %s", nextagel.arr);
  switch (temptabl) (
     ----
        EXCEC SQL OFFER CAllO;
        if (sqlos.sqloods) {
          SLOUTS ("CURSOR CAllo");
           ENDONSG;
           SLOVER ("4s", sqlca.sqlerm.sqlermc);
        for (;;) {
          EXEC SQL FETCE CAllO into :camb;
           if (sqloa.sqloods - SQL_BOF) breek;
          if (sqloa.sqloods) {
             SLOUTP ("FETCE CALLO");
             MEDCAMO;
             SLOUTE ("4s", sqloa.sqlerm.sqlermc);
          EIRC SQL SELECT entry num
                     FROM qual_cit1
                     23720 : £00
                    WHERE entry_num = :enumb;
          if (sqlos.sqloods = SQL_EOF) {
             EXEC SQL DEEDT DETO QUAL CITA (entry aum) VALUES (:enumb);
             if (sqloa.sqloods) {
                SLOUIS ("IMSERT Table 1");
                REPORTED;
                SLOUTS ("%s", sqloa.sqlarma.sqlarma);
             #1fdef CENCROTE
               enumb.arr[10] = '\0';
               printf("\nts qualifies", enumb.arr);
            EXEC SQL COMMET WORK;
             qual_estries++;
            MENVALS ();
      EXCEC SQL CLOSE CA110;
    0856 1:
      EXCEC SQL OFER CAll1;
```

```
if (sqlos.sqloods) (
               SLOUTP ("CURSOR CAll1");
               SLOUIF ("4s", sqloa.sqlarm.sqlarms);
           for (;;) {
              MANG SQL FETCH CAll! into :esumb;
              if (sqlos.sqloode = SQL BOF) break;
if (sqlos.sqloode) {
    sLOOTP ("FETCH CAlll");
                  ENDOMAS;
                  SLOTTP ("ts", sqlom.sqlerrm.sqlerrmc);
              EXEC SQL SELECT entry_num
                          FROM qual_cit2
                          2070 :foo
                         WEEKE entry_num = :enumb;
              if (sqlos.sqloods = SQL_EOF) (
                  EXEC SQL DESERT DETO QUAL_CITY (entry_arm) VALUES (:enumb);
                  if (sqlos.sqloods) {
                     BEDOMEG;
                     SLOUTS ("4s", sqloa.sqlerm.sqlerma);
                 #1fdef CHRCHOUT
                    enumb.arr[10] = '\0';
                    printf("\n0s qualifies", enumb.arr);
                 EXEC SQL COMMET WORK;
                 qual_estries++;
                 HERVALS ();
          MOUNT SQL CLOSE CA111;
           break:
       GR40 2:
          MINISC BOL OFFER CAll2:
           if (sqlos.sqloode) {
              BEDOMES:
              SLOUTE ("CORSOR CAll2");
              SLOUTS ("4s", sqloa.sqlorms.sqlorms);
           for (;;) {
              MIEC SQL FETCE CAll? into :enumb;
              if (sqloa.sqloods == SQL BOF) break; if (sqloa.sqloods) {
                 SLOOTS ("FERCE CAll2");
                 EMPORAGE.
                 SLOUTP ("4s", sqlca.sqlermm.sqlermmc);
              EXEC SQL SELECT entry hum
                         FROM qual citi
                         13770 : foo
                        WHERE entry_num = :enumb;
             if (sqloa.sqloods - SQL_ROF) (
EXEC SQL ISSERT INTO QUAL CITI (entry num) VALUES (:enumb);
                 if (sqlca.sqlcode) {
                    SLOURS ("4s", sqlos.sqlerm.sqlermo);
                 #1fdef CRECKOUT
                    emumb.arr[10] = '\0';
                   printf("\nts qualifies", enumb.err);
                 MINEC SQL COMMIT WORK;
                qual_entries++;
MENVALS();
                )
          EDEC SQL CLOSE CA112;
         break;
      default:
         bad_temp();
   EXEC SQL FETCE CA100 INTO :nexternl;
   } while (!sqlca.sqlcode);
if (sqlos.sqloods := sqt_Bor) {
EMDCHSG;
   SLOUTS ("FETCE CA100");
   SLOUTE ("4s", sqlca.sqlerum.sqlerumc);
```

```
temptabl = (temptabl + 2) + 1;
 EDEC SOL CLOSE CA100;
 Biffed CHICKOUT
    printf(" %d entries in table %d", qual_entries, temptabl);
 rotura (int) qual_estrice;
 queryhl.pc -- Routines for ditation database retrieval in the
                      human area using keyword set 1
       This file contains:
 /* Beader for calls to MS-DOS
 #include cocess.h>
                                                                             */
 #inalude <stdio.b>
 #define SQLCA_STORAGE_CLASS extern
                                       /* Switch for header files
 EXEC SQL REGIN DECLARE SECTION;
 EXCEC SOL TECHNOL cityare.h:
 EXEC SOL ESD DECLARS SECTION:
 EXEC SOL INCLUDE SOLCA;
 finclude 'esen.h'
                                        /* Standard ASAW Header File
 static char far *hoursors[] = {
 "SHURCT entry num FROM headquarters ditation search WHERE h assovanc = 'T'".
 "SELECT q.estry_num FROM headquarters.citation_search s, qual_cit1 q\
 MEERS s.estry_num = q.entry_num AND s.h_ennoyenc = 'T'
 "SELECT q.estry_num FROM headquarters.citation_search s, qual_cit2 q\
MRERS s.estry_num = q.estry_num AND s.h_annoyanc = 'T'",
 "SELECT entry_num FROM headquarters.citation_search NEERE h_psychley = 'T'",
 "SELECT q.estry_num FROM headquarters.citation_search s, qual_cit1 q\
WEEDE s.estry_num = q.estry_num AND s.h_psychlgy = 'T'",
 *SELECT q.entry_num FROM headquarters.citation_search s, qual_cit2 q\
 WHERE s.entry_num = q.entry_num AND s.h_psychloy = 'T'".
"SELECT estry_num FROM headquarters.citation_search NEERE h_physical = 'T'",
*SELECT q.estry_num FROM beadquarters.citation_search s, qual_cit1 q\
 WHERE s.estry_num = q.estry_num AND s.h_physical = 'T'",
"SELECT q.entry_num FROM headquarters.citation_search s, qual_cit2 q\
 WHERE s.entry_num = q.entry_num AND s.h_physical = 'T'",
"SELECT entry num FROM headquarters, mitation search WEERE h sleep = 'T'".
"SELECT q.estry_num FROM headquarters.ditation_search s, qual_dit1 q/
 WHERE s.entry_num = q.entry_num AND s.h_sleep = 'T'"
"SELECT q.estry_num FROM headquarters.citation_search s, qual_cit2 q\
WEERE s.estry_num = q.estry_num AHD s.h_sleep = "T'",
"SELECT entry num FROM headquarters ditation search WHERE h speech = 'T'",
"SELECT q.entry_num FROM headquarters.citation_search s, qual_cit1 q\
 WHERE s.entry num = q.entry num AED s.h_speech = 'T'"
"SELECT q.entry_num FROM headquarters.ditation_search s, qual_dit2 q\ WHERE s.entry_num = q.entry_num AND s.h_speech = 'Z'",
"FELECT entry num FRCM headquarters.citation_search WEERE h perfranc = 'T'",
"SELECT q.entry_num FROM headquarters.citation_search s, qual_cit1 q\
WHERE s.entry_num = q.entry_num AND s.h_perfranc = '7'",
"SKLECT q.entry_num FROM beadquarters.citation_search s, qual_cit2 q\
WHERE s.entry_num = q.entry_num AND s.h_perfranc = 'T'");
```

```
int hoursoffeet:
ist queryhi()
         .
    queryhl -- Create the subset of citations for the titles that
            pertain to the NUMNN area and also select the affect
            at level 1
#15def CERCEOUS
 printf("\nqueryhl ");
Bear 11 4
SLOUT ("Ruman Area") ;
switch (eff2srch[0]) {
  case 0: break:
  case 1: queryhl1(); breek;
  case 2: queryh12(); breek;
  case 3: queryh13(); breek;
  case 4: quaryh14(); break;
  case 5: queryh15(); breek;
  case 6: queryh16(); break;
#1fdef CERCEDUT
  default: sprintf(workspace.arr, "Invalid Ruman Rffect 4d", eff2srdh[0]);
        SLOUTS (workspace.arr) ;
#end1£
retura (ist) qual_estries;
int queryhil()
queryhil -- Create the subset of citations for the titles that
          pertain to the EUMAN area and also ANNOYANCE
#154af CHBCHOUT
 printf("\mqueryhll ");
Pendif
stroat (selcrit, "Ruman Ann");
hoursoffset = 0;
return (int) queryhlz();
queryhi2 -- Create the subset of citations for the titles that
            pertain to the EURAN area and also PSYCHOLOGICAL MEALWH
fifdef CHECKOOT
 printf("\nqueryh12 ");
Sendi#
street (selerit, "Dsychol");
hoursoffset = 3:
return (int) queryhiz();
int queryhil()
   queryhl3 -- Create the subset of citations for the titles that
            pertain to the HUMAN area and also PHYSICAL HEALTH
```

#1fdef CRECEDUT

Seeds 2

printf("\nqueryh13 ");

```
stroat (selerit, "Thys. Seelth");
hoursoffset = 6;
return (int) queryhlm();
ist queryhl4()
/****************************
    quaryhl4 -- Create the subset of citations for the titles that
             pertain to the EURCH area and also SLEEP INTERFERENCE
**********************************
#1fdef CENCEOUT
  printf("\mqueryhi4 ");
Seedif.
struct (selarit, "Eunex Sleep");
hoursoffset = 9;
return (int) queryhlz();
int queryh15()
queryhi5 -- Create the subset of citations for the titles that
             partain to the NORM area and also SPEECE INTERFERENCE
**********************
#1fdaf CEBCEOUT
  printf("\aqueryh15 ");
stroat (selerit, "Speech");
hoursoffset = 12;
return (int) queryhlm();
int queryh16()
    quaryhi6 -- Create the subset of ditations for the titles that
            pertain to the NUMBER area and also TASK PERFORMANCE
#1fdef CENCROOT
  printf("\mqmeryh16 ");
Pendif
street (selerit, "Euman Perf");
hoursoffset = 18;
return (int) queryhlz();
int queryhix()
    queryhiz -- Actually do the search using hourseroffset
#1fdef CEECHOUT
 printf(" quaryhlz %d+%d ", hoursoffset, temptabl);
Seedif
stropy(sqlstmat.arr, hoursors[hoursoffset+temptabl]);
sqistmat.lea = strlen(sqistmat.arr);
EXEC SQL PREPARE D1 FROM :sqlstmat;
if (sqlca.sqlcode) (
  ENDOMSG;
  SLOUTS ("Frepare CED");
  SLOUTP ("%s", sqloa.sqlorms.sqlorms);
  amit (16);
```

```
EXEC SOL DECLARS CED CURSOR FOR DI-
 MINE SQL OFFER CHO;
 if (sqlca.sqlcode) {
    ELOUTE ("Open CED");
    SLOUTS ("4s", sqlos.sqlorms.sqlorms);
 qual_estries = 0;
 for (;;) {
    EXEC SQL FERCH CHD into :enumb;
    if (sqlos.sqloods - SQL 205) break;
    if (sqlos.sqloods) {
      HIDOMS;
      SLOUTS ("Total CED");
      SLOUTS ("4s", sqlox.sqlorms.sqlorms);
   if (temptabl == 1) {
    EXEC SQL INSERT INTO QUAL_CIT2 (entry_num) VALUES (:enumb);
      EXEC SQL IMPERT INTO QUAL_CIT1 (entry_aum) VALUES (:enumb);
   if (sqlos.sqloods) {
       SLOUTS ("Insert after CED");
       SLOUTP ("ts", sqloa.sqlarm.sqlarma);
       exit (16);
   qual entries++;
   MENVALS () ;
   EXEC SQL COMMETT WORK;
MINIC SQL CLOSE CMD;
temptabl = (temptabl 4 2) + 1;
#1fdef CERCEOUT
   printf(" %d extrice in table %d", qual_extrice, temptabl);
 retura (ist) qual_estrice;
queryh2.ps -- Routines for mitation database retrieval in the
                   human area using keyword set 2
      This file contains:
finalude oprocess.b>
                                     /* Header for calls to MS-DOS
finalude (stdio.b)
#define SQLCA_STORAGE_CLASS extern /* Switch for header files
MINC SQL BRGIN DECLARS SECTION;
MING SQL INCLUDE cityers.h;
EXEC SQL END DECLARE SECTION;
EXEC SQL INCLUDE SQLCA;
                                     /* Standard ASAW Seeder File
static char far *h2cursors[] = {
"SKLECT d.entry_num FROM headquarters.citation_details d,\
headquarters.citation_search s WHENE d.entry_num = s.entry_num/
AND d.aircraft = 'T'",
"SELECT q.entry_num FROM headquarters.citation_details d, qual_cit1 q,\
headquarters.citation_search s WEERE s.entry_num = q.entry_num\
AND d.entry_num = q.entry_num AND d.eircraft = '7'",
"SELECT q.estry_num FROM headquarters.citation_details d, qual_cit2 q,\
```

}

- headquarters.citation\_search s WHERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.eircraft = 'T's,
- "SELECT d.entry num FROM headquarters.ditation\_details d,\
  headquarters.ditation\_search s WEERE d.entry\_num = s.entry\_num\
  d.ns\_blast = '2'",
- "SELECT q.entry\_num FROM bendquarters.citation\_details d, qual\_cit1 q,\
  headquarters.citation\_search s WEERS s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.ms\_blast = 'T'",
- "FELECT q.entry\_num FROM headquarters.citation\_details d, qual\_cit2 q,\
  headquarters.citation\_search s WEERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.as\_blast = 'T'",
- "SELECT entry\_num FROM headquarters.citation\_details d,\
  headquarters.citation\_search s WEEKE d.entry\_num = s.entry\_num\
  d.as\_seismic = '7'",
- "SELECT q.entry num FROM headquarters.citation\_details d, qual\_citl q,\
  headquarters.citation\_search s WEERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.ns\_seismin = '7'",
- "SHLECT q.entry num FROM headquarters.citation\_details d, qual\_cit2 q,\
  headquarters.citation\_search s NHERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.ns\_seismic = '7'",
- "SELECT entry num FROM headquarters.citation\_details d,\
  headquarters.citation\_search s WHERE d.entry\_num = s.estry\_num\
  d.ms\_searchm = 'T'".
- "SELECT q.estry\_num FROM headquarters.citation\_details d, qual\_cit1 q,\
  headquarters.citation\_search s WEERS s.entry\_num = q.estry\_num\
  AND d.estry\_num = q.estry\_num AND d.as\_socion = 'T'",
- "SHLECT q.entry\_num FROM headquarters.citation\_details d, qual\_cit2 q,\
  headquarters.citation\_search s WHERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.ms\_sonichm = '7'",
- "SELECT d.entry num 'FROM headquarters.citation details d,\
  headquarters.citation\_search s WEERE d.entry\_num = s.entry\_num\
  AND d.ns\_terrain = 'T'",
- "SELECT quantry num FROM headquarters.citation\_details\_d, qual\_citl\_q,\
  headquarters.citation\_search\_s\_WEERE\_s.extry\_num = q.estry\_num\
  AND\_d.estry\_num = q.estry\_num\_AND\_d.as\_terrain = "F"",
- "SELECT q.entry\_num FROM beadquarters.citation\_details d, qual\_cit2 q,\
  beadquarters.citation\_search s WEERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.as\_terrain = 'T'",
- "SELECT d.entry num FROM headquarters.citation\_details d,\
  headquarters.citation\_search s NEERE d.entry\_num = s.entry\_num\
  d.traffic = '7'",
- "SELECT q.entry\_num FROM headquarters.citation\_details d, qual\_cit1 q,\
  headquarters.citation\_search s WHERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.traffic = 'T'",
- "SELECT q.entry num FROM headquarters.citation\_details d, qual\_cit2 q,\
  headquarters.citation\_search s WHERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.traffic = 'T'',
- "SELECT entry\_num FROM headquarters.citation\_details d,\
  headquarters.citation\_search s WEERE d.entry\_num = s.entry\_num\
  d.wind\_nse := 'T'",
- "SELECT q.entry\_num FROM headquarters.citation\_details d, qual\_cit1 q,\
  headquarters.citation\_search s WEERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.wind\_nse = 'T'",
- "SELECT q.entry\_num FROM headquarters.citation\_details d, qual\_cit2 q,\
  headquarters.citation\_search s WEERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.wind\_nse = "7",
- "SELECT entry\_num FROM headquarters.citation\_details d,\
  headquarters.citation\_search s WHERE d.entry\_num = s.entry\_num\
  d.othr nse = '7'",
- "MELECT q.eatry\_num FROM headquarters.citation\_details d, qual\_cit1 q,\
  headquarters.citation\_search s WEERE s.entry\_num = q.entry\_num\
  AND d.entry\_num = q.entry\_num AND d.othr\_nse = "T'",

```
"SKLECT q.entry_num FROM headquarters.ditation_details d, qual_dit2 q,\
headquarters.ditation_search s NEERS s.entry_num = q.entry_num\
  AND deserty new = questry new AND desthr nee = 'T'"};
  int cursoroffset;
 int queryh2()
  queryh2 -- Create the subset of citations for the titles that
               pertain to the NUMBH area and also select the effect
               at level 2
 ************************************
 #1fdef CHICKNOT
   printf("\mqueryh2 ");
 SLOUT ("Rumen Area");
 switch (officerch[1]) {
   case 0: breek;
    case 1: queryh21(); break;
   case 2: queryh22(); break;
   case 3: queryh23(); break;
   case 4: queryh24(); break;
case 5: queryh25(); break;
   case 6: queryh26(); break;
   case 7: queryh27(); break;
   case 8: queryh28(); break;
 Bifdef CHRCHOUT
   default: sprintf(workspace.arr, "Invalid Ruman Effect &d", eff2srch[1]);
          SLOUIS (werkspace.arr);
 feedif
 return (int) qual_estries;
 int queryh21()
 queryh21 -- Create the subset of mitations for the titles that
          pertain to the NUMBH area and also Aircraft Noise
#1fdef CERCEOUT
  printf("\mqueryh21 ");
streat (selecit, "Aircraft");
Garsoroffset = 0;
return (int) queryh2x();
1st queryh22()
queryh22 -- Create the subset of citations for the titles that
          pertain to the EURAF area and also Blast Hoise
fifdef CHECKUT
 printf("\nqueryh22 ");
street (selerit, "Blast");
oursoroffset = 3;
return (ist) queryh2x();
```

```
int queryh23()
    quaryh23 -- Create the subset of citations for the titles that
       pertain to the EURAH area and also Seismin
**********************************
#1fdef CHICKOUT
  printf("\mqueryh23 ");
feadif
street (selerit, "Seismie");
cursoroffset = 6;
retura (ist) queryh2x();
ist queryh24()
quaryh24 -- Create the subset of citations for the titles that
    pertain to the NUMNN area and also Sould Boom
#1fdef CHECKOUT
  printf("\nqueryh24 ");
dead15
street (selerit, "Boom") ;
cursoroffset = 9;
return (int) queryh2z();
int quaryh25()
   quaryh25 -- Create the subset of ditations for the titles that
           pertain to the EURIN area and also Terrain
    Revision Eistory:
    1.00 02/04/88 ahr
                      1. Creation
ifdef CERCEOUT
 printf("\nqueryh25 ");
feedif
street (selerit, "Terrein");
cursoroffset = 12;
return (ist) queryh2x();
int quaryh26()
   queryh26 -- Create the subset of citations for the titles that
           pertain to the FOROW area and also Traffic
$15def CENCHOUT
 printf("\nqueryh26 ");
Seed1.5
streat(selerit, "Traffie");
cursoroffset = 15;
return (int) queryh2x();
ist queryh27()
queryh27 -- Create the subset of citations for the titles that
           pertain to the NUMBN area and also Wind Noise
```

```
Sifdef CEBCEOUT
  printf("\nqueryh27 ");
Beadle
struct (selerit, "Wind");
cursoroffset = 18;
return (int) queryhiz();
queryh28 -- Create the subset of ditations for the titles that
         partain to the EURON area and also Other Noise
#1fdef CERCEOUT
  printf("\aqueryh28 ");
Sendiff.
street (selerit, "Other");
cursoroffset = 21;
return (int) queryh2x();
ist queryh2z()
    queryhiz -- Actually do the search using hoursoroffset
#1fdef CHBCBOOT
 printf(" queryhiz 4d+4d ", oursoroffset, temptabl);
stropy(sqlstant.arr, h2cursors(cursoroffset+temptabl));
sqlstant.lea = strlea(sqlstant.arr);
EXEC SQL PREPARE D1 FROM :sqlstmat;
if (sqlam.sqloods) {
  INDOMS;
  SLOUTS ("Frepare CED") ;
  SLOUTS ("%s", sqlos.sqlorms.sqlorms);
  exit (16);
MANC SQL DECLARE CED CURSOR FOR D1:
MICHE SQL OFFER CHO;
if (sqlos.sqloods) {
  EMDOMEG:
  SLOUTS ("Open CED");
SLOUTS ("$s", sqloa.sqlarms.sqlarms);
qual_estries = 0;
for (;;) {
  EDDIC SQL FETCE CED into :enumb;
  if (sqlca.sqlcode == SQL_BOF) breek;
  if (sqlos.sqloods) {
    SLOOTS ("Fetch CED");
    SLOUTP ("4s", sqlox.sqlerms.sqlerms);
 if (temptabl = 1) {
    EXEC SQL INSERT INTO QUAL_CIT2 (entry_num) VALUES (:enumb);
    EXEC SQL INSERT INTO QUAL_CIT1 (entry_aum) VALUES (:enumb);
  if (sqlca.sqlcode) (
     ENDOMS:
     SLOUTS ("Insert after CED");
```

```
SLOOTS ("4s", sqlom.sqlorms.sqlorms);
        exit (16) ;
    qual estries++;
    MING SQL COMMET WORK;
    HEWVALS ();
EDGEC SOL CLOSE CED:
 tomptabl = (tomptabl + 2) + 1;
 #1fdef CENCEDUT
   printf(" %d entries in table %d", qual_entries, temptabl);
 return (int) qual_entries;
 queryh3.pc -- Nowtines for citation detabase retrieval in the
                     human area using keyword set 3
       This file contains:
 *******************************
Sinclude corocess.b>
                                        /* Header for calls to MM-DOS
Singlade <stdio.k>
Medine SQLCA STORAGE CLASS extern
                                       /* Switch for header files
MING SOL MEGIN DECLARE SECTION;
EXEC SQL INCLUDE citvare.h;
EXEC SOL END DECLARE ENCRICE:
EXEC SQL INCLUDE SQLCA;
dinglade "sess.h"
                                        /* Standard ASAN Reeder File
static char far *hloursors[] = {
"SELECT d.entry_num FROM headquarters.citation_details d,\
 headquarters.citation_search s NEERE d.eatry_num = s.eatry_num/
 AND d.field_expt = 'T'",
*SELECT q.entry_num FROM beadquarters.ditation_details d, qual_dit1 q,\
 And destry arm = questry arm AND destry arm = questry arm \
"SELECT q.entry_num FROM beadquarters.citation_details d, qual_cit2 q,\
 headquarters ditation search s WEERS steatry num = q.estry num\
AND d.entry_num = q.entry_num AND d.field_expt = 'Z'",
"SELECT d.entry_num FROM headquarters.citation_details d,\
 headquarters.citation_search s MERE d.entry_num = s.entry_num\
 d.lab_ampmat = 'T'",
"SKLECT q.estry_num FROM headquarters.citation_details d, qual_cit1 q,\
headquarters.citation search s WEERE s.eatry num = q.eatry num \
AND d.eatry num = q.eatry num AND d.lab expent = 'T'',
"SELECT q.entry_num FROM headquarters.nitation_details d, qual_nit2 q,\
 headquarters.citation_search s WEERE s.entry_num = q.entry_num/
 AND d.entry_num = q.entry_num AND d.lab_expent = 'T'",
"SELECT entry num FROM beadquarters ditation details d,\
beadquarters ditation_search s WHERE d.entry_num = s.entry_num\
 review art = 'T'",
"SKLECT q.entry_num FROM headquarters.citation_details d, qual_cit1 q,\
beadquarters.ditation_search s WHERE s.entry_num = q.entry_num\
AND d.entry_num = q.entry_num AND d.review_art = '2'",
"SELECT q.entry_num FROM beadquarters.citation_details d, qual_cit2 q,\
headquarters ditation search s NEECS s.estry.num = q.estry.num\
AND d.estry.num = q.estry.num AND d.review.art = 'Z'',
"SELECT entry_num FROM headquarters.citation_details d,\
 beedquarters ditation search s WHERE d.entry num = s.entry num/
proposl_ar = 'T'*,
"SELECT q.estry num FROM headquarters.citation details d, qual cit1 q,\
```

```
AND d.entry_num = q.entry_num AND d.proposl_nr = 'T'",
"SELECT q.entry_num FROM beadquarters.citation_details d, qual_cit2 q,\
beadquarters.citation_search s NEERS s.entry_num = q.entry_num\
AND d.entry aum = q.entry aum AND d.proposl ar = 'T'");
int cursoroffset;
int queryhi()
quaryhl -- Create the subset of citations for the titles that
pertain to the NAGAM area and also select the effect
             at level 3
#1fdef CENCEDUT
  printf("\nqueryh3 ");
Seed15
SLOUT ("Eumen Aree") ;
switch (eff2srch[2]) {
  case 0: breek;
  case 1: queryh31(); breek;
  case 2: quaryh32(); break;
  case 3: queryh33(); breek;
case 4: quaryh34(); break;
#ifdef CEBCEOUT
  default: sprintf(workspace.arr, "Invalid Euman Effect %d", eff2srch[2]);
         SLOUTS (workspace.arr);
fondif
return (int) qual_entries;
ist queryh31()
queryhll -- Create the subset of aitations for the titles that
         pertain to the MMAN area and also Field Experiment
#1fdef CENCROTE
 printf("\nqueryh31 ");
stroat(selcrit, "Euman Fld");
cursoroffset = 0;
return (int) queryhiz();
ist queryh32()
queryh32 -- Create the subset of citations for the titles that
         pertain to the EURCH area and also leb experiment
************************
Bifdef CENCEDUT
 printf("\mqueryh32 ");
feed1.f
stroat (selcrit, "Euman Lab");
Gursoroffset = 3;
return (ist) queryh3x();
ist querykl3()
```

headquarters.citation\_search s NEERE s.entry\_num = q.entry\_num\

```
quaryhl3 -- Create the subset of citations for the titles that
             pertain to the EURAN area and also Review Article
*************************
#1fdef CEBCECOTE
  printf("\nqueryh33 ");
-
stroat(selcrit, "Euman Rov");
cursoroffset = 6;
return (int) queryhlz();
ist queryh34()
queryhl4 -- Create the subset of mitations for the titles that
             pertain to the EURCH area and also Theoretical
#1fdef CHBCROUT
  printf("\mqueryh34 ");
Bendif
stroat (selcrit, "Euman Theor");
cursoroffset = 9;
return (int) queryhlz();
int queryblz()
queryhlz -- Actually do the search using hoursoroffeet
#15def CEBCHOOT
  printf(" queryhlz 4d+4d ", oursoroffset, temptabl);
stropy(sqlstmat.arr, hloursors[oursoroffset+temptabl]);
sqlstmat.len = strlen(sqlstmat.arr);
EXEC SQL PREPARE D1 FROM :sqlstant;
if (sqlos.sqloods) {
  SLOUTP ("Frepare CED");
  SLOUTP ("%s", sqloa.sqlarm.sqlarmc);
  exit (16) ;
EXEC SQL DECLARE CED CURSOR FOR D1;
EXCEC SOL OFFER CED:
if (sqloa.sqloods) {
  SLOUTP ("Open CED");
  SLOUIF ("4s", sqlos.sqlerim.sqlerimo);
qual entries = 0;
for (;;) (
  EXEC SQL FETCH CHD into :eaunb;
  if (sqloa.sqloods == SQL_BOF) break;
  if (sqloa.sqloods) {
    EMEDICANS :
    SLOUTP ("Fetch CED");
    SLOUTS ("4s", sqloa.sqlorma.sqlormo);
  if (temptabl == 1) (
EXEC SQL INNER! INTO QUAL_CIT2 (entry_num) VALUES (:enumb);
  ele (
    EXEC SQL IMSERT INTO QUAL_CIT1 (entry_num) VALUES (:enumb);
```

```
if (sqlos.sqloods) {
        ENDOMEG;
        SLOUTP ("Insert after CED");
        SLOUTP ("4s", sqlos.sqlerms.sqlerms);
        exit (16);
   qual_estrics++;
   HENVALE ();
EDEBC SQL CLOSE CED;
temptabl = (temptabl + 2) + 1;
Bifdef CENCEOUT
  printf(" %d entries in table %d", qual_estries, temptabl);
 Seedi £
return (int) qual_entries;
quarypo.po -- Routines for point of contact retrieval
      This file contains:
findlude      
                                      /* Header for calls to MS-DOS
finelnde <stdio.b>
finclude <ctype.h>
#define SQLCA_STORMS_CLASS extern /* Switch for header files
EXEC SQL BEGIN DECLARS SECTION;
EXEC SOL INCLUDE cityars.h:
VARCEAR crit1[16];
VARCEAR crit2[46];
VARCEAR crit3[18];
VARCEAR orit4[10];
VARCHAR crit5[26];
MINE SOL MED DECLARE SECTION:
EMEC SQL DECLUDE SQLCA:
finclude "sess.h"
                                       /* Standard ASAN Header File
static int poc_type;
EXEC SOL DECLARE POCI CURSOR FOR
                 SELECT first name, last name, title, office, agray dept, st add div, po box, misc add, city base, state, miscode, mail code, phone, affiliation,
                         maj_attrib, min_attrib,
                         area, scope_auth
                  FACM superuser point of contact
WHERE UPPER(last_name) LIEE :crit1
AND UPPER(min_attrib) LIEE :crit2
                     AND UPPER (maj_sttrib) LIRE : crit3
                    AND UPPER (affiliation) LIKE : crit4
                    AND UPPER(state) = :crit5;
MING SQL DECLARE POC2 CURSOR FOR
                 SELECT first name, last name, title, office,
                        agnoy_dept, st_add_div, po_box,
                        misc add, city base, state, zipcode,
                        mail code, phone, affiliation,
                        maj attrib, min attrib,
                        area, scope_auth
                  FROM superuser.point of contact
WHERE UPPER(last_name) LIME :crit1
AND UPPER(min_sttrib) LIME :crit2
                    AND UPPER (maj attrib) LIKE : crit3
                    AND UPPER (affiliation) LIKE : crit4
                    AND UPPER (city_base) LIKE :crit5;
```

```
postoh -- Set up the search in the point of contact detabase
 18t 1, 5;
static char *affil[7] = ("CITY",
                           "COUNTY",
                          "SEATES",
                          "PEDERALA",
                          "MILITARYS",
                          "TRIBAL4",
                          -4-1:
#1fdef CESCROUT
   printf("\mposroh ");
 feadif
SLOUP("Foint of contact search");
stropy(crit4.arr, affil[whichafl]);
crit4.len = strlen(crit4.arr);
j = strlem(contaem);
for (i = 0; i < j; i++) orit1.arr[i] = toupper(contamn[i]);
orit1.arr[j] = '\0';</pre>
stroat (crit1.arr, "+");
crit1.len = strlen(crit1.arr);
j = strlem(minoret);
for (i = 0; i < j; i++) grit2.arr[i] = toupper(minorat[i]);
grit2.arr[j] = '\0';</pre>
stroat (grit2.arr, "+");
crit2.len = strlen(crit2.arr);
j = strles(mejorst);
for (i = 0; i < j; i++) arit3.arr[i] = toupper(majoret[i]);
arit3.arr[j] = '\0';
strost(arit3.arr, "4");</pre>
crit3.lea = strlea(crit3.arr);
j = strlem(address);
for (i = 0; i < j; i++) arit5.arr[i] = toupper(address[i]);
arit5.arr[j] = '\0';</pre>
crit5.len = strlen(crit5.arr);
1£ () = 2)
   poc_type = 1;
also (
 pod_type = 2;
stroat(orit5.arr, "%");
  crit5.lem++;
if (pod_type == 1) {
    EXEC SQL OPEN POC1;
   if (sqlos.sqloods) {
      SLOUIS ("Cursor POC1");
      estipOnseg () ;
      exit (16);
      }
alsa {
   MARC SQL OPEN DOCE;
   if (sqlma.sqlmode) {
      SLOTTP ("Cursor POC2");
      emponeg ();
      exit (16);
1f (astpos()) {
  HER_SCREEN ("contactscreen");
   SLOUTS ("Sorry, but I cannot find anybody like that...");}
also (
  NEW_SCREEN("poodisplay");
  }
```

ist posrch()

```
int autpos()
  ampos -- Find next instance in the point of contact detabase
  switch (poc_type) (
        EXEC SQL FERGE POCI DETO :f name, :l_name, :contitle, :office,
                                                              :agency_dept, :st_add_div, :po_box,
                                                              :misc_add, :dity_base, :state, :misc_add,
                                                               :mail code, :phone, :affiliatio,
                                                               :major_attrib, :mimor_attribute,
                                                              :Bres, :#00pe;
       if (sqlos.sqloods) {
              if (sqlos.sqloods - SQL_BOF) {
                    pod type = 0;
EXEC SQL CLOSE FOCL;
                    SLOUTS ("No more entries!");
                    return (int) SQL_BOF; }
               else {
                   SLOUIP ("Fetch POC1");
                     empOmeg();
                    exit (16) :
       EXEC SQL FETCH POC2 INTO :f name, :l name, :contitle, :cffice, :agency_dept, :st_add_div, :po_box, :misc_add, :dity_base, :state, :misc_add, :dity_base, :misc_add, :misc_add, :dity_base, :misc_add, :m
                                                              :mail_mode, :phome, :affiliatio,
                                                              :major_attrib, :mimor_attribute,
                                                             :area, :scope;
       if (sqlos.sqloods) {
              if (sqlos.sqloods - SQL_NOF) (
                   pod_type = 0;
                     MINE SQL GLOSE POCE;
                    SLOUTP ("No more entries!");
                    return (int) SQL_BOF; }
              alse (
                    SLOUZP ("Fetch FOC2");
                    emponing ();
                    emit (16);
       break:
        GREG 0:
                   SLOURS ("There REALLY are no more entries!");
                    return (int) 0;
 f_nme.arr[f_nme.len] = '\0';
 l name.arr[l name.len] = '\0';
contitle.arr[contitle.len] = '\0';
 office.arr[office.len] = '\0';
 agency_dept.arr[agency_dept.lem] = '\0';
st_add_div.arr[st_add_div.len] = '\0';
po_box.arr[po_box.len] = '\0';
 misc_add.arr[misc_add.lea] = '\0';
city base.arr[city base.lea] = '\0';
state.arr[state.lea] = '\0';
 zipoode.arr[zipoode.lem] = '\0';
mail_code.arr[mail_code.lea] = '\0';
phone.arr[phone.len] = '\0';
 affiliatio.arr[affiliatio.len] = '\0';
major_sttrib.arr[major_sttrib.lem] = '\0';
minor_sttribute.arr[minor_sttribute.lem] = '\0';
area.arr[area.len] = '\0';
scope.arr[scope.len] = '\0';
MENVALS ();
return (int) sqlom.sqloode;
int setaff(1)
```

```
autpor -- Find next instance in the point of contact database
int i:
static *affects[] = ("CITY", "COUSTY", "STATE", "FREERAL", "MILITARY",
                 "TRIBAL", "NOME");
whichafl = 1;
stropy(affselos, affsels[1]);
MENVALE();
querys1.pc -- Routines for citation database retrieval in the
                structures area using keyword set 1
     This file contains:
/* Reader for calls to MS-DOS
#define SQLCA_STORAGE_CLASS extern /* Switch for header files
EXEC SOL REGIS DECLARS SECTION:
REEC SQL INCLUDE ditvers.h;
EXEC SQL END DECLARE SECTION:
MINEC SQL INCLUDE SQLCA:
                              /* Standard ASAN Reeder File
int querysi()
Alfdet CHRCHOUT
  printf("\aquaryel ");
feedif
SLOUTEP ("Structures Area Unavailable");
return (int) qual_estries;
searches.pd -- Routines that use the query routines to do searches
    This file contains:
    heroh001 - Search based on entries on first Ruman area screen
    asrch001 - Search based on entries on first Animal area screen
    sarch001 - Search based on entries on first Structures area screen
    merch001 - Search based on entries on first Modeling area screen
    shwartcit - Display the next citation on the screen
    loadscit - Load the next citation from the database FOR DISPLAY
finalude   finalude <stdio.b>
                             /* Header for calls to MS-DOS
#define SQLCA_STORAGE_CLASS extern
                             /* Switch for header files
EXEC SQL REGIS DECLARE SECTION;
                             /* All SQL declarations are in */
EXEC SQL INCLODE sitvers.h;
EXEC SQL END DECLARS SECTION;
EXEC SQL THULDDE SQLCA;
finclude "asan.h"
                             /* Standard ASAN Reader File
/*********************
                        SQL CRESCRE
```

```
MESC SQL DECLARE CD00 CURSOR FOR
        SELECY authornum FROM headquarters.author gitation link
        WEEKE entry_num = :enumb;
 MESC SQL DECLARS CD10 CURSOR FOR
        SELECT q.entry num, s.suitability, t.title, s.date pub
FROM headquarters.ditation_titles t,
              headquarters.citation_search s,
        quel_cit1 q
          AND s.estry_num = q.estry_num;
 MINE SQL DECLARE CD20 CURSOR FOR
        SELECT q.entry_num, s.suitability, t.title, s.date_pub
        FROM beadquarters citation titles t,
              headquarters.citation_search s,
        qual_cit2 q
          AND s.estry_num = q.estry_num;
 Screen "Memory"
 int herehood()
 WHEREAMI (Screen, Window, Datum, Sutton);
stropy(oldscreen, Screen);
selerit[0] = '\0';
 int asrahooo()
 MEEREAMI (Screen, Window, Datum, Button);
 stropy (oldscreen, Screen);
 selarit[0] = '\0';
int serohooo()
 MEEREAMI (Screen, Window, Detum, Button);
stropy (oldscreen, Screen);
selerit[0] = '\0';
int march000()
WHEREAMI (Screen, Window, Datum, Button);
stropy(oldscreen, Screen);
selerit[0] = '\0';
int herch001()
hardh001 - Search based on entries on first Euman area some
#1fdef CHBCROUT
printf("\mberch001 ");
herah000();
1f (queryw()) {
  RENVALS();
  if (queryd()) (
       MERVALS ();
       dsexAp () :
    )
  }
HEWVALS ();
if (|qual_entries) {
  SLOUTEF ("No entries matching these criteria were found");
  return (int) 0;}
else {
  if (|malisto()) {
   if (|shwartait()) {
```

```
MMW_SCREEK("citdspl");
           ADD_WINDOW("citdispastica", 19, 1);)
     return (int) 0;}
  int hershooz ()
        herch002 - Search based on entries on Ruman area keys
  ********************************
  printf("\nherah002 ");
  -
  if (queryw()) {
     MENVALS ();
     1f (queryt()) {
        HENVALS ();
        if (queryd()) {
           MENVALE ();
           1f (queryh1()) {
              if (queryh2()) {
                 queryh3 ();
 MENVALS ();
 if ([qual_entries) (
    SLOUTER ("No entries matching these criteria were found");
    return (int) 0;}
 alsa (
    if (!malisto()) {
   if (!shwartait()) {
          HEW_SCREEN ("attdspl");
          ADD_WINDOW("citdispaction", 19, 1);}
    return (int) 0;}
 int asrch001()
       asrch001 - Search based on entries on first Aminal area scre
#1fdef CHECKOUT
printf("\mesrch001 ");
Pendis
if (queryw()) {
    memvals();
   if (queryt()) (
      MERVALS ();
      12 (queryd()) {
    menvals();
         destls () :
      }
HENVALS ();
if (iqual_entries) {
    sicovery("No entries metching these criteris were found");
   return (int) 0;}
alsa {
 if (|molisto()) {
     if (!shwartait()) {
    Hem_screen("citdspl");
         ADD WINDOW("citdispaction", 19, 1);}
  return (int) 0;}
```

```
int serch001()
       ssrch001 - Search based on entries on first Structur
  Bifdef CENCEDUZ
 printf("\n seroh001 ");
  feadif
 1f (queryw()) {
    HENVALS ();
    if (queryt()) (
      MENVALS ();
      if (queryd()) {
    mawvals();
         querys ();
   }
 MENVALS ();
 if ([qual_estries) {
    SLOUTEF("No entries metching these criteria were found");
    return (int) 0; )
 also {
   if (!malisto()) (
    if (!shwaxtait()) {
        MEM_SCHEM("citdspl");
AND_WINDOW("citdispartics", 19, 1);}
   return (int) 0;}
 int merch001()
             - Search based on entries on first Modeling area sore
 ************************************
 #1fdef CERCHOUT
 printf("\merch001 ");
 Bends#
if (queryd()) {
        MENVALS();
        querym ();
HEWVALS();
return (int) 0;}
alse {
  if (|mulisto()) {
     if (!shwastait()) {
       MEN_SCREEN("citdspl");
       ADD_WINDOW("citdispection", 19, 1);}
  return (int) 0;}
imt molisto()
    melisto - Open the cursor for main citation display scree
*************************************
```

```
#1fdef CENCHOUR
     printf("\mmalisto ");
   curreit = 0;
   switch (temptabl) (
     COSC 1:
EXEC SQL OPEN CD10;
        1f (sqlos.sqloods) {
sLOUTS("Cursor CD10");
          emponeg();
          exit (16);}
       break;
     case 2:
       MEET SQL 0998 0020:
       if (sqlos.sqloods) (
          ELOUIS ("Cursor CD20");
          exposes ();
          exit (16);}
       breek;
     default:
       bed_temp(temptabl);
  return (int) sqlom.sqloods;
  int malista()
       meliste - Close the cursor for main citation display screen
  #1fdef CENCROUT
    printf("\mmalista ");
 #endif
 switch (temptabl) (
    case 1:
      EXEC SQL CLOSE CD10;
      1f (sqlos.sqloods) (
sLoors("Cursor CD10");
         emponeg ();
         exit (16);}
      breek:
      EXCEC SQL CLOSE CD20;
      if (sqlos.sqloods) (
        SLOUTS ("Cursor CD20");
        emponeg();
        exit (16);}
   default:
     bed_temp(temptabl);
return (int) sqlom.sqloods;
int shwartoit()
shwarteit - Show next citation
***********************************
int select;
#ifdef CEECHOUT
  printf("\nshwmatcit ");
Sendi!
if (select = loadscit()) {
```

```
switch (select) (
        -
        CREASE SOL FETCE OFF OF CHORRE
           MLOUTER ("You are at the end of the list");
           REMOVE_WINDOW();
           breek:
        dofault :
           exit (16);
       1
  also {
     curreit++;
     MENVALS ();
  retura (int) sqloa.sqloods;
       loadscit - Load the citation detail display with the next citation
 int 1, 3, k, m;
 #1fdef CENCEDOT
    printf("\mloadmoit ");
 SLOOT ("Retrieving Display Text");
 if (temptabl == 1) {
    EDEC SQL FETCH CD10 INTO :enumb, :suitable, :workspace, :datep;
    if (sqloa.sqloods) return (int) sqloa.sqloods;
 else {
   EXEC SQL FETCE CD20 INTO :enumb, :suitable, :workspace, :datep;
    if (sqloa.sqloods) return (int) sqloa.sqloods;
   1
    mb.arr(enumb.lem) = '\0';
 suitable.arr[1] = '\0';
 workspace.arr[workspace.lea] = '\0';
 datep.arr[datep.len] = '\0';
 details_read = 0;
j = (int) (((float) workspace.len) / ((float) 60)) + 1;
j = j > 4 ? 4 : j;
k = 0;
for (1 = 0; 1 < j; 1++) {
    n = k + 60 > workspace.len ? workspace.len - k : 60;
   strncpy(entdesc[i].arr, &workspace.arr[60*i], n);
   estdess[1].les = 2;
   entdesc[1].arr[n] m '\0';
   k += a;
for ( ; i < 4;i++) entdesc[i].arr[0] = '\0';</pre>
MARC SQL 0970 CD00;
if (sqlos.sqloods) (
  SLOUZE ("Cursor CD00");
   amponeg();}
k = 0:
for (;;) {
  v40point = Sauthorlist[k];
   authorlist[k].len = 40;
/* for (i= 0; i < 40; i++) authorlist[k].arr[i] = ' '; */
  EXEC SQL FETCE CD00 INTO :authornum;
  if (sqlom.sqloods) break;
  REEC SQL SELECT author FROM headquarters author list
       DETO : v40point WHERE authornum = :authornum;
  if (sqloa.sqloods) {
     MLOUTE ("Author selection...");
     emponing ();
```

```
authorlist[k].arr[authorlist[k].les] = '\0';
     2++;
  12 (sqlos.sqloods != SQL_BOF) {
     if (sqlos.sqloods) {
       ELCOTE ("Cursor CDOO");
        exponeg();
       exit (16);}
     else SLOOTP ("Mart citation: more authors than screen shows");
  EXCEC SQL CLOSE COO;
 for (i = k; i < 10; i++) {     /* WULL for those not filled this time */
     swthorlist[i].err[0] = '\0';
     swthorlist[i].len = 0;}
</pre>
  12 (k > 4)
                            /* Truncate if we have more than 5 */
    for (1 = 0; 1 < k; 1++) {
       authorlist[k].lea = 1 > 20 7 30 : 1;
       authorlist[k].arr[authorlist[k].lea] = '\0';}
  return (int) 0;
  ist reservab()
  research - Rescope the search. If no citations, clear the tables.
 ***********************************
 if (!quel_entries) quetup();
MEW_SCREEN(olderress);
 int hereh010()
      herch010 - Setup for entries on Euman keys
 printf("\nherch010 ");
 Seedif.
hereh000();
if ([qual_entries) (aff2srch[0] = aff2srch[1] = aff2srch[2] = 0;}
NEW_SCREEN ("hkeyerch");
int herch011()
     herch011 - Ruman Effect Descriptor Type Varification
#1fdef CHRCHOTZ
printf("\nherch011 ");
كنهمه
for(;;) {
  if (desctype.arr[0] = 'A')
if (desctype.arr[0] = 'P')
                                              {eff2srch[0] = 1; break;}
      15
      if (desctype.arr[1] = 's')
else if (desctype.arr[1] = 'z')
                                              {eff2srch[0] = 2; break;}
                                              {aff2srch[0] = 3; break;}
  if (desctype.arr[0] = 's')
if (desctype.arr[1] = 'L')
                                              {aff2srch[0] = 4; break;}
      also if (desctype.arr[1] = 'p')
                                              {eff2srch[0] = 5; break;}
```

emit (16);}

```
if (desctype.arr[0] = 'T')
                                                     {eff2srch[0] = 6; breek;}
     if ((eff2srch[0]) || (desctype.arr[0] == ' ')) (eff2srch[0] = 0; break;}
     MOUTE ("Not a valid effect");
     UPDATE_DATCM("desctype");
  ADD_WINDOW("moistype", 7, 24);
WEET_DATOM("moistype");
  int heroh012()
        herdh012 - Hoise Descriptor Type Verification
  fifdet CENCEDOT
  printf("\mherch012 ");
  -
  for(;;) {
    if (desctype.arr[0] == 'A')
if (desctype.arr[0] == 'B')
if (desctype.arr[0] == 'B')
                                                    {eff2srch[1] = 1; break;}
                                                    (eff2srch[1] = 2; break;)
        if (desctype.arr[1] == 'R')
else if (desctype.arr[1] == '0')
        12
                                                    {eff2srch[1] = 3; breek;}
{eff2srch[1] = 4; breek;}
    if (desctype.arr[0] == 'T')
if (desctype.arr[1] == 'E')
                                                    {eff2srch[1] = 5; breek;}
        else if (desctype.arr[1] = 'R')
                                                    {eff2srch[1] = 6; break;}
    if (descrippe.arr[0] = 'W')
if (descrippe.arr[0] = 'O')
                                                    {aff2srch[1] = 7; break;}
                                                    {eff2srch[1] = 8; break;}
    if ((eff2srch[0]) || (desctype.arr[0] == ' ')) {eff2srch[1] = 0; breek;}
st.corm("Not a valid noise type");
    UPDATE_DATUM("moletype");
 REMOVE_WINDOW();
 ADD_WINDOW("exprtype", 7, 24);
MENT_DATOM("expdess");
 heroh013 - Experimental Type Varification
 *************************************
#1fdef CHECKOTT
printf("\mbsrch013 ");
 fead1f
for(;;) {
  Af (desctype.arr[0] == 'F')
                                                   {eff2srch[2] = 1; breek;}
   if (desctype.arr[0] == 'L')
                                                   {aff2srch[2] = 2; break;}
   if (desctype.arr[0] == 'R')
                                                   {eff2srch[2] = 3; breek;}
   if (desctype.arr[0] = 'T')
                                                   {aff2srch[2] = 4; break;}
   if ((eff2srch[0]) || (desctype.arr[0] = '')) {eff2srch[2] = 0; breek;}
SLOUTS("Not valid type");
   UPDATE_DATUM("exprtype");
REMOVE_WINDOW();
int serch010()
      serok010 - Setup for entries on Structures keyword screen
```

```
printf("\mssrch010 ");
fendif
   eszak000 ();
  if ([qual_entries) {eff2srch[0] = eff2srch[1] = eff2srch[2] = 0;}
  MEN_SCREEN("skeysrch");
  int merch010()
  march010 - Setup for entries on Modeling keyword screen
  #1fdef CHBCROVE
  printf("\mmerch010 ");
  feedif
 merch000();
 if ([qual_entries) (eff2sroh[0] = eff2sroh[1] = eff2sroh[2] = 0;}
 MEN_SCHEEN ("Bleysrob");
 int esrch010()
 asrch010 - Setup for entries on Animal Reyword scre
 #1fdef CENCROTY
 printf("\nasrch010 ");
 Sand12
 esrah000();
 species1.arr[0] = '\0'; species1.len = 0;
species2.arr[0] = '\0'; species2.len = 0;
species3.arr[0] = '\0'; species3.len = 0;
species4.arr[0] = '\0'; species4.len = 0;
MRWALS();
 MER_SCREEN ("akeyerch");
 textdisplps -- Routines that display text fields from "MMNO_FILE"
    This file contains:
#include cprocess.b>
                           /* Reader for calls to ME-DOS
finalude (stdio.b)
$define SQLCA_STORAGE_CLASS extern /* Switch for header files
                                                    •/
EXEC SQL REGIN DECLARS SECTION;
                           /* All SQL declarations are in
EXEC SQL DECLODE cityars.h;
MORE SQL MED DECLARE SECTION;
EXEC SQL INCLUDE SQLCA;
finciade "asan.h"
                           /* Standard ASAW Header File
int deplahet()
```

```
deplahet -- Display abstract
  {
  #1fdef CENCHOUT
    printf("\n deplahet ");
  Sendic.
  if (|details_read) get_details();
  if (abstract.les) {
   stracpy(maximumo.arr, abstract.arr, 10);
    getmeno () ;
    MENT_SCREEN("shownhe");
ADD_WINDOW("shwantabe", 21, 1);}
    SLOUTP ("No abstract available");
    }
 ist deplorit()
 deplorit -- Display critique
 fifdef CENCROTT
   printf("\m deplorit ");
 if (!details_read) get_details();
 switch (critdisp) (
      if (critiquel.len = 10) (
         stracpy (nextmeno.arr, critiquel.arr, 10);
         getaeso ();
         HEN SCHEEK ("showabs");
         ADD_WINDOW("shwantrev", 21, 1);
         MENVALE ();
         critdisp = 1;
         return (int) critdisp; )
case 1:
      if (critique2.len == 10) {
    strncpy(nextmemo.arr, critique2.arr, 10);
        getneso () ;
        NEW_SCREEN ("showabs");
        ADD_WINDOW("shwartrey", 21, 1);
        MENVALS ();
        critdisp = 2;
        return (int) critdisp; }
Case 2:
     if (critique).les == 10) {
        stracpy (nextmeno.arr, critique3.arr, 10);
        getmeno () ;
        MEM_SCHEM("showabs");
ADD_WINDOW("shwartrev", 21, 1);
        MERVALS ();
        critcisp = 3;
        return (int) critcisp; }
default:
    SLOUTS ("There are no more critical reviews");
    critdisp = 0;
ADD_WINDOW("shwaxtabe", 21, 1);
    return (int) critdisp;
```

```
getmano -- Retrive a memofield from the detabase
  FILE *fopen(), *memo;
  int fwrite();
  #1fdef CEECHOUT
   printf(" getmeno ");
  See 11.5
  mextmeno.len = 10;
  EXEC SQL SELECT memo_text FROM headquarters.memo_file 1870 :bigdisplay
                  WEEKE block number w :nextmemo;
  if (sqloa.sqloods) {
   SLOUTS ("Cannot find memo block");
    SLOUTS (sqlca.sqlarma.sqlarma);
   return (int) 0;)
 return (int) 0;}
 fwrite(bigdisplay.arm, bigdisplay.lem, 1, memo);
fprintf(memo, "\m\n");
fclose(memo);
 return (int) 0;
 }
 get_details -- get the pointers to the detailed memo fields
 #1fdef CERCEOUT
  printf(" getdetails ");
 Send15
 EXEC SQL SELECT RESTRACT, CRITIQUE_R1, CRITIQUE_R2, CRITIQUE_R3
       FROM headquarters distation details WHERE entry num = : cause
       DETO :abstract, :critiquel, :critiquel, :critiquel;
if (sqlos.sqloods) {
  SLOUTS ("Cannot find details");
  EDEDOMAG -
  SLOUTS (sqlos.sqlerrs.sqlerrsc);
  return (int) 0;}
#1fdef CENCEDOT
  printf(" Abs %d, Crit %d, %d, %d", abstract.len, critiquel.len,
         critique2.len, critique3.len);
critdisp = 0;
details_read = 1;
... - ma* - All user information for a particular MTR
#define SQLCA STORAGE CLASS extern /* Switch for header files
```

```
MINIC SQL INCLUDE SQLCA;
  EXEC SQL REGIS DECLARS SECTION;
                                      /* All SQL declarations are in */
   EIEC SQL INCLUDE hostwars.h;
                                      /* these header files
  ENEC SQL INCLOSE Scharris.h;
  HERC SQL BED DECLARS SECTION;
  finalude 'asan,h'
                                      /* Standard ASAN Header File
  int melisto()
  melisto --- Open cursor $2 for a list of user information
                        of navigation points on a particular MTR
       Routine executes an open cursor command for cursor $2 and then
       returns to the calling program with the CRACLE status code.
       Note: Modifications to this function may impact the related
               functions mtlistf() and mtlistc() that fotch rows and
               close the cursor and, possibly, functions that call
               these willity routines.
  #1fdef CERCHOUT
  printf("\mmalisto ");
  Sendif.
  stropy( aid.arr, sraid.arr);
  stroat (aid.arr, "4");
  did.len = strlen(did.arr);
 MINEC SQL DECLARE S2 CURSOR FOR
         FELECT fix label, floor ref, cailing ref, fix id, fix type, artor, fix lat, fix lon, fix red, fix dist, floor, cailing, width laft, width right
                  mtroegments
                  fix label LIEE : gid
          ORDER BY fix label;
 MINC SQL OFFER $2;
 fifdef CESCHOUZ
 printf(" Open: %ld ", sqlom.sqloods);
 Bearing.
 retura (int) sqlos.sqloods;
 int malists()
 malistf --- Fetch a row using the opened cursor $2 for MTR
                     Mavigation Point User Information
      Routine amountes an fetch command for oursor $2, which is assume
      to have been opened, and them returns to the calling program with
      the ORACLE status code
      Note: Modifications to this function may impact the related
             functions mtlisto() and mtlisto() that open and close the
             oursor and, likely, functions that call these utilities
stropy(optr3, *
                         "); /* If you don't use VARCHAR, you have to */
"); /* clear the space or waird things happen */
stropy(optr4, "
EXEC SQL FERCE S2 1970 :82bv,
                     :a2bv, :optrl, :optr2,
:ourfixtyp, :ourarted, :optr3,
:ourfixed, :ourfixdist, :lptr1,
                                                          : curfixid,
                                                          : optr4,
                                                          :lptr2,
                     : curwidleft, : curwidright;
prefixed.arr[prefixed.lea] = '\0';
```

```
prefixtyp.arr[prefixtyp.lea] = '\0';
  #1fdef CENCEDOTE
  printf(" Fetak: %ld ", eqloa.eqloode);
  Sends?
  return (int) sqlos.sqloode;
  ist malista()
  /**********************************
            malista --- Close cursor S2 for Mavigation Points
      Boutine executes a close cursor command for cursor $2 and then
      returns to the calling progrem with the ORACLE status code.
      Note: Modifications to this function may impact the related
              functions mtlisto() and mtlistf() that open the cursor
             and fetch rows using it and, possibly, functions that
             call these stilities
 #1fdef CHRCHOTZ
 printf("\matlista ");
 Seeds 2
 EXEC SOL CLOSE S2:
 #1fdef CENCROOT
 printf("Close: tld ", sqlos.sqloods);
 return (int) sqlos.sqloods;
 /*
                  ASAN REPORT GENERATOR MODULE
 /•
 /*
/*
                       Jenuary 26, 1987
 finalude  process.b> finalude <stdio.b>
                                  /* Header for calls to MS-DOS
 #define SQLCA_STORAGE_CLASS extern
                                  /* Switch for header files
EXEC SQL REGIN DECLARS SECTION;
                                  /* All SQL declarations are in */
EXEC SQL INCLUDE hostware.h:
EXEC SQL INCLUDE fabarris.h;
VARCEAR comparemis[8];
VARCEAR comparemit[26];
                                  /* Last mission used */
                                  /* Last MER used */
VARCHAR comparesc[12];
                                  /* Last aircraft used */
double exptab[27];
                                  /* Space for "ROUTE.BAS" Table */
int day_ops[12], might_ops[12];
ROOM SQL MED DECLARE SECTION;
EDEC SQL DECLODE SQLCA;
finciade "esen.h"
                                 /* Standard ASAN Seader File
EXEC SQL DECLARE LINEAL CURSOR FOR
                     SELECT s_label, m_ident, mircraft, activity
                      TROM activities
                   ORDER BY s_label, siroreft, s_ident;
MINC SQL DECLARS LINEAR CURSOR FOR
                    SELECT day, might FROM operations
                   NEEDE activity = :activity
ORDER BY month ASC;
EXEC SQL DECLARE LINDAS CORSOR FOR
                     SELECT s_label, m_ident, siroraft, activity
                      FROM activities
                   ORDER BY s label, m ident, siroraft;
```

```
char
  int
  int
          annoy flag = -2, hearing flag = -2;
sleep flag = -2, livestock flag = -2, math[20];
  int
 int speech flag = -2, glass flag = -2; index; int compare flag = -2, species flag = -2, index; double ldmr1, ldmr2, ldmr3, ldmr4, ldmr5, ldmr[20][5];
 double leq1, leq2, leq3, aleval[20][5];
 report (zame)
   er ame[];
    FILE
            *fopea(), *zpt;
    ist
    stropy(altlev," ");

rpt = fopen(name, "w");

if (rpt = NULL) {
       printf("\nRed filename to", name);
       exit(16);}
    table1(rpt);
    tbl2(spt);
    table3 (zpt);
    EXEC SQL OFFE LIBOAL;
    if (sqlox.sqloods) {
  printf("Open 3: %s", sqlox.sqlerms,sqlerms);
       exit(4); }
    for (1=0;;1++) (
       BURG SQL FETCE LIEDAS DETO :sreid, :misslabl, :ac name, :activity;
       if (sqlos.sqloods - SQL_ROF) break;
       if (sqlos.sqloods) {
          printf("Fetch 3: %s", sqlcm.sqlermc);
          exit(4); }
       srcid.arr[srcid.lem] = '\0';
       missiabl.arr(missiabl.lem) = '\0';
       ad_neme.arr[ad_neme.len] = '\0';
       fillrtavals(i);
       landuse (rpt, 1);
       inconsequential (rpt, 1);
       minor (rpt, 1, 1);
       considerable (rpt, 1, 1);
       motocomsidered(rpt,1);
   EXCEC SQL CLOSE LINEAL;
   references (rpt) ;
   folose (spt);
report3 (name, which)
char nmo[];
int which;
(
   FILE *fopen(), *rpt;
           1:
   stropy(altley, " ");
   rpt = fopen(name, "w");
if (rpt == MOLL) {
      printf("\nRed filenume 4s", nume);
      exit (16);}
   ENGIC SQL OFFER LINDAY;
   if (sqlos.sqloods) {
      printf("Open 3: %s", sqlox.sqlorms.sqlorms);
      exit(4); }
   for (1=0;;1++) {
     EXEC SQL FETCE LIEDAS INTO :sreid, :misslabl, :ac asso, :activity;
      if (sqloa.sqloods - sql_Bor) break;
     if (sqlca.sqlcode) {
   printf("Fetch 3: %s", sqlca.sqlerma.sqlerma);
         exit(4); }
      srcid.arr[srcid.len] = '\0';
     misslabl.arr[misslabl.lem] = '\0';
     ad_name.arr[ad_name.len] = '\0';
     filirtavals (1) ;
     landuse (spt, 1);
     inconsequential (rpt, i);
```

```
minor (rpt, i, which);
          considerable (rpt, 1, which);
          motocomidered(rpt,1);
      EXEC SQL CLOSE LINEAR;
      references (rpt) ;
      falose (zpt);
   table1(zpt)
  FILE *zpt;
      optr3 = show.let;
      optr4 = show.lon;
      comparentr.arr[0] = '\0';
      BODEC SQL OFFER LIMBAL;
      if (sqlos.sqloods) {
         printf("Cursor 1: 4s", sqlom.sqlerm.sqlerma);
         exit (4);
      fprintf(rpt, "DESCRIPTION OF PROPOSED ACTION(n/n");
      fprint(;pt,"The proposed action, known as %s,\n",ASSESMONT.name);
fprintf(xpt,"(%s)\n",ASSESMONT.desc);
      fprintf(rpt, "consists of the use of the MTR segments as described in Table \
  fprintf(xpt, "The subscale flight operations proposed for these MTR segments\n");
fprintf(xpt, "are described in Table 2.\n\n");
     for (;;) {
        MING SQL PETCH LIMBAL INTO :sreid, :misslabl, :ad_memo, :activity;
         if (sqlos.sqloods - sql_mor) break;
         if (sqloa.sqloods) (
           printf("Fetch 1: %s", sqlox.sqlerms.sqlerms);
            exit(4): 1
        if (!stracep(comparentr.arr, sraid.arr, sraid.len)) continue;
        strapy(comparatr.arr, sreid.arr, sreid.len);
sreid.arr[sreid.len] = '\0';
        missiabl.arr[missiabl.len] = '\0';
ac_name.arr[sc_name.len] = '\0';
   fprintf(rpt, "Table 1: Description of MTR %s\n\n", sraid.arr);
  fprintf(xpt," MAV \n");
fprintf(xpt," FOIRT FIX RAD/DIS
                                                      LATITODE
                                                                          LONGITUDE\n"1:
   fprintf(mpt, "-----
                                                                           ----\z-);
        if (malisto()) {
           exposes ();
           exit (4);
    1
        while ( |malistf() ) (
           {int 1, 5;
           for (imercid.len, j=0; i < m2bv.len; i++, j++)
           curnavpt.arr[j] = a2bv.arr[i];
curnavpt.arr[j] = '\0';}
           carfixid.arr[curfixid.lea] = '\0';
       show.let[3] = ' ';
       show.lon[3] = ' ';
fprintf(rpt, " %s
                                  %# $03d/$03d
                                                                    46\2",
                                                           40
       curnevpt.arr, curfixed.arr, curfixed, curfixdist, show.lat, show.lon);
    fprintf(rpt, "\n\n\n");
   if (sqloa.sqloods != SQL_BOF) printf("\aks", sqloa.sqlorm.sqlormo);
   malista();
EXEC SQL CLOSE LIMBAL;
tb12(rpt)
FILE *rpt;
   int i;
```

```
comparentr.arr[0] = '\0';
      COMPARAGE .AFF[0] = '\0';
       EXCEC SQL OFFE LINDAL;
      if (sqlox.sqloods) {
         printf("Cursor 1: 4s", sqlom.sqlarms;;
         exit(4);}
      for (::) {
         EXEC SQL FETCH LIMBAL INTO :sreid, :misslabl, :ad_name, :activity;
         if (sqlos.sqloods == SQL_BOF) break;
         if (sqlom.sqloods) {
   printf("Fetch 1: *s", sqlom.sqlerrm.sqlerrmd);
            exit(4); }
         if (stracep(comparentr.arr, sraid.arr, sraid.lea)) {
            sraid.arr[sraid.len] = '\0';
            stracpy(comparentr.arr, srcid.arr, srcid.les);
        if (stracep(comperenc.arr, ad_name.arr, (unsigned int)12)) {
           sc name.arr[sc name.len] = '\0';
stracpy(comparesc.arr, sc name.arr, (unsigned int) 12);
           EXEC SQL SELECT power_units FROM streeltab
                            2070 :pr_pwr_u WEERS aircraft = :as_assa;
           if (sqloa.sqloods) (
              printf("Fower units %s", sqlos.sqlerms.sqlerms);
               exit (4);}
           pr_pur_u.arr[pr_pur_u.lea] = '\0';
        misslabl.arr[misslabl.len] = '\0';
        EXEC SQL SELECT SORTISSIES FROM RESSIONS INTO : AC in form
                                   WHERE mission = :misslabl;
        if (sqlca.sqlcode) {
           printf("Sortie size ts", sqlom.sqlerma.sqlerma);
           exit (4);}
        EXEC SQL SELECT alt, alt_ref, per, spd
                        FROM mtr_flight_pers
                      DEFO :althoral, :alther, :ac power, :ac speed
                        AMD seq = 1;
        if (sqlos.sqloods) (
          printf("Flight permeters ts", sqlca.sqlerm.sqlerme);
           if (sqlos.sqloods == SQL_ROF) { /* For obsolving only | */
              altlevel = 300;
               stropy(altley, "? *");
               ac_power = 100.00;
              ac_speed = 450;
          alse exit(4);
       fprintf(rpt, "\n\n\nTable 2: Description of Flight Operations on %s \
 by Month", sroid.arm);
       fprintf(rpt, "\n\n\mission %s (%d AIRCRAFT/FORMATION)\n\n",
                  misslabl.arr, ac in form);
       fprintf(rpt, "AIRCRAFT: 4s
                                                    PORER: 47.215 46\2",
       sc_name.arr, sc_power, pr_per u.arr);
fprintf(rpt, "ALTITODE: $6d ts spec
                                                    SPEED: 47.21f hts\m\m",
                  altievel, altiev, ac_speed);
       fprintf(rpt, "
                                               OPERATIONS\n");
       fprintf(rpt, "MONTE
                                       DAY
                                                              MIGET\R" } :
       fprintf(rpt, -----
                                                             ----\n");
      REEC SQL OFFER LINDA2;
      if (sqlos.sqloods) {
         printf("Cursor 2: %s", sqlom.sqlermm.sqlermmo);
         exit (4); }
      EXEC SQL FETCE LIBDAY ISTO :day_ops, :might_ops;
      for (1=0; 1 < 12; 1++) {
         fprintf(rpt, * 4s
                                         144
                                                                $4d\x*,
                month[i], day_ops[i], might_ops[i]);}
      ENEC SOL CLOSE LINDAZ;
   EXEC SQL CLOSE LIMBAL;
table3 (zpt)
```

```
FILE *mpt;
1
        1, 1;
double temp1, temp2, add, log10();
fprintf(rpt,"\fsummar of PREDICTED HOISE EXPOSURE\a\a");
fprintf(rpt, "Noise exposure produced by miroraft operations may be \
specified in a variety\n");
fprintf(rpt, "of waits. The noise of low altitude high speed flights on \
Military Training\a*);
Sprintf(xpt, "Noutes is specified for current purposes by a communitive moise \
metric called\n");
fprintf(rpt, "the caset rate adjusted monthly day-night average, A-weighted \
sound level, \a");
fprintf(rpt, "abbreviated Ldmmr. As described by Flotkin et al. (1987), \
this metric is\n");
fprintf(rpt, "based on an integration period equal to the calendar menth \
with the highest\n");
fprintf(rpt, "number of operations.\n\n\n");
comparements.arr[0] = '\0';
EXEC SQL OFFER LINEAS;
if (eqlos.eqloode) {
   printf("Cursor 3: ts", sqloa.sqlerm.sqlerma);
   exit (4);
fprintf(xpt, "Table 3: Summary of Maximum Hoise Exposure Produced by Flight \
Mission Aircraft Month Distance
                                                                      Ldmar\n\n*);
\=");
for (j=0;;j++) {
    EURC SQL FETCE LUDAS INTO :srcid, :misslabl, :mc_mema, :activity;
   if (sqlos.sqloods — SQL_BOF) break;
   if (sqlos.sqloods) (
      printf("Fetch 3: %s", sqlox.sqlerms.sqlerms);
      exit(4): }
   sraid.arr[sraid.lea] = '\0';
   if (stracep(comparements.arr, srcid.arr, srcid.len)) (
      stracpy(comparemetr.arr, srcid.arr, srcid.lea);
      fprintf(xpt, "\a4-9s", srcid.arr);
      comparemis.arr[0] = '\0'; /* insure that these fail for new m/c */
      comparesc.arr[0] = '\0';
   else fprintf(rpt, "\m
  misslabl.arr[misslabl.lem] = '\0';
  if (stracep(comparemis.arr, misslabl.arr, misslabl.len)) {
     stracpy(comparemis.arr, misslabl.arr, misslabl.len);
comparemis.arr[comparemis.len] = '\0';
comparems.arr[0] = '\0';
     MINC SQL SELECT sortie size FROM missions INTO :ac_in_form
                                 WEERE mission = :misslabl;
     if (sqlca.sqlcode) {
        printf("Sortie size %s", sqlcs.sqlerrm.sqlerrmd);
        azit (4);}
     fprintf(rpt, "4-9s", misslabl.arr);
  else fprintf(rpt, "
  ac_neme.arr[sc_neme.len] = '\0';
  if (stracep(comperedc.arr, ac name.arr, ac name.lea)) {
     stracpy(comparesc.arr, ac name.arr, ac name.len);
     comparesd.arr[ad_neme.len] = '\0';
fprintf(rpt, " %-9s", ad_neme.arr);)
se fprintf(rpt, "\n ");
  else fprintf(rpt, \a
  EXEC SOL OFER LINDA2:
  if (sqlos.sqloods) {
     printf("Cursor 2: activity %d\n %s", activity, sqlca.sqlerms.sqlermsc);
     exit (4); }
  EXEC SQL FETCE LIMBA2 INTO :day_ops, :might_ops;
  if (sqlos.sqloods) {
     printf("Fetch 2: 4s", sqlcs.sqlerms.sqlermc);
     exit (4); }
```

```
MINC SQL CLOSE LINDA2;
  add = 0;
  temp2 = 300.0;
  for (1=0; 1 < 12; 1++) {
     temp1 = (double) (day_ope[i] + 10 * might_ope[i]);
if (temp1 > add) {
        if (temp1 < temp2) temp2 = temp1 ;
        add = temp1;
        index = 1;
  if ((add-temp2) < 0.05) index = 12;
 math[j] = index;
 EXEC SQL SELECT sortie size
                  FROM missions
                DETO :ad in form
 if (sqloe.sqloods) (
    printf("Sortie size ts", sqlca.sqlarm.sqlarmc);
     exit (4);}
 if (add > 0) add = 10.0 * log10(add * (double) ac_in_form) - 64.1;
 else add = -1000.00;
    /* This is a bit sloppy we should subtract 10 LOGIO of
       the number of seconds in the month instead of 64.1, but .... */
 sprintf(workspace.arr, "sminC? ACT+04d FROM mtr_emp_tab", sotivity);
 workspace.len = strlem(workspace.arr);
 EXEC SQL PREPARE STAT FROM : Workspace;
 if (sqloa.sqloods) {
    printf("Frepare: %s", sqlos.sqlerm.sqlermo);
    exit(16);}
 EXEC SQL DECLARE D_CURS CURSOR FOR STATE;
 ROLL SQL OFFER D CURE;
 if (sqlca.sqlcode) {
    printf("Dyn Open: %s", sqlom.sqlerrm.sqlerrmd);
    exit (16) ; )
 BEES SQL FETCE D_CURS INTO :emptab;
 if (sqlos.sqloods) {
    printf("Exposure table: ts", sqlom.sqlerm.sqlerme);
    if ((sqlos.sqloods == SQL_BOF) ||
(sqlos.sqloods == NULL_FETCHED))
                                             /* For checking only ! */
       for (i = 0; i < 27; i++) exptab[i] = 0.0;
     also axit(4);
 EXEC SQL CLOSE D COME;
1f (add > -500.0) {
   ldmar1 = amptab[0] + add;
   alevel[j][0] = exptab[0];
alsa {
   ldmr1 = (double) 0;
   alevel[j][0] = 0;
12 (add > -500.0) {
   ldmar2 = emptab[16] + add;
   aleval[j][1] = amptab[16];
alse {
   ldmax2 = (double) 0;
   alevel[]][1] = 0;
1f (add > -500.0) {
  ldmar3 = amptab[19] + add;
aloval[j][2] = amptab[19];
} معلم
  ldmr3 = (double) 0;
  alevel[j][2] = 0;
if (mdd > -500.0) {
   ldmr4 = 0.5*(exptab[20]+exptab[21])+edd;
```

```
alevel[j][3] = 0.5*(exptab[20]+exptab[21]);
            mr4 = (double) 0;
        alevel[j][3] = 0;
    if (mdd > -500.0) {
        ldmr5 = emptab[22] + edd;
        mlevel[j][4] = emptab[22];
    oleo {
       14
           mar5 = (double) 0;
       alevel[j][4] = 0;
    ldmr[j][0] = ldmr1;
    ldmr[j][1] = ldmr2;
    ldmr[j][2] = ldmr3;
    ldmr[j][3] = ldmr4;
    ldomr[j][4] = ldomr5;
    fprintf(rpt, " 4-3s
                            0.0 miles
                                               ", month[index]);
    if (ldmr1 >= 45.0) fprintf(xpt, "46.11f", ldmr1);
                    else fprimtf(rpt, " -");
    Sprintf(rpt, "\a\t\t
                                                  0.5 miles
                                                                     -) ;
    if (ldmar2 >= 45.0) fprintf(rpt, "46.11f", ldmar2);
                    alse fprintf(rpt," -");
    fpristf(rpt, "\a\t\t
                                                  1.0 miles
                                                                     -);
    if (ldmar3 >= 45.0) fprintf(rpt, "%6.11f", ldmar3);
                    else fprimtf(mpt, " -");
    fpristf(rpt, "\a\t\t
   else fprintf(rpt, " -");
   fprintf(rpt, "\a\t\t
                                                                     -);
   if (ldmar5 >= 45.0) fprintf(rpt, "46.11f", ldmar5);
                    else fprintf(rpt, " -");
EXEC SQL CLOSE LIBERS;
fillriavals (1)
int 1;
   rta_vals1[1][0] = annoyance (ldmar[1][0], &(rta_vals1[1][1]));
   rta_vals2[1][0] = annoyance (ldmar[1][1], &(rta_vals2[1][1]));
  rtn_vals2[1][0] = annoyance (ldmar[1][2], &(rtn_vals2[1][1]));
rtn_vals4[1][0] = annoyance (ldmar[1][3], &(rtn_vals4[1][1]));
   rtn_vals5[1][0] = manoyance (ldmar[1][4], &(rtn_vals5[1][1]));
   rtn_vals1[2][0] = hearing_demage (alevel[i][0], & (rtn_vals1[2][1]));
  rts_vals2[2][0] = hearing_demage (aleval[1][1], &(rts_vals2[2][1]));
rts_vals3[2][0] = hearing_demage (aleval[1][2], &(rts_vals3[2][1]));
  rtn_vels4[2][0] = hearing_demage (alevel[1][3], &(rtn_vels4[2][1]));
   rtm_vals5[2][0] = hearing_demage_(alevel[i][4], & (rtm_vals5[2][1]));
  rtn_vels1[3][0] = sleep_interference (alevel[i][0], &(rtn_vels1[3][1]), i);
rtn_vels2[3][0] = sleep_interference (alevel[i][1], &(rtn_vels2[3][1]), i);
  rtn_vals3[3][0] = sleep_interference (alevel[i][2], &(rtn_vals3[3][1]), i);
   rtn_vals4[3][0] = sleep_interference (alevel[i][3], &(rtn_vals4[3][1]), i);
  rtm_vals5[3][0] = sleep_interference (alevel[4][4], &(rtm_vals5[3][1]), 1);
  rta_vals1[4][0] = sen_lead_use (4(rta_vals1[4][1]));
  rts_vels2[4][0] = see_land_use (6(rts_vels2[4][1]));
  rts_vals3[4][0] = sen_land_use (4(rts_vals3[4][1]));
rts_vals4[4][0] = sen_land_use (4(rts_vals4[4][1]));
  rtn_vals5[4][0] = sem_land_use (&(utn_vals5[4][1]));
  rtn_vals1[5][0] = endangered_species_reproduction (&(rtn_vals1[5][1]));
  rtn_vals2[5][0] = endangered_species_reproduction (&(rtn_vals2[5][1]));
 rts_vals3[5][0] = endangared species_reproduction (&(rts_vals3[5][1]));
rts_vals4[5][0] = endangared_species_reproduction (&(rts_vals4[5][1]));
  rts_vals5[5][0] = andangared_species_reproduction (&(rts_vals5[5][1]));
 rtn_vals1[6][0] = livestock_dumg (&(rtn_vals1[6][1]));
 rtm_vals2[6][0] = livestock_demg (&(rtm_vals2[6][1]));
 rts_vals3[6][0] = livestock_damg (4(rts_vals3[6][1]));
rts_vals4[6][0] = livestock_damg (6(rts_vals4[6][1]));
```

```
rta_vals5[6][0] = livestock_demg (&(rta vals5[6][1]));
    rtn_vals1[7][0] = speech_interference (&(rtn_vals1[7][1]));
    rts_vals2[7][0] = speech_interference (&(rts_vals2[7][1]));
rts_vals2[7][0] = speech_interference (&(rts_vals2[7][1]));
rts_vals4[7][0] = speech_interference (&(rts_vals4[7][1]));
     rts_vals5[7][0] = speech_interference (&(rts_vals5[7][1]));
     rta_vals1[8][0] = glass_breakage (&(rta_vals1[8][1]));
     rta_vals2[8][0] = glass_breakage (&(rta_vals2[8][1]));
    rtn_vals3[8][0] = glass_breakage (6(rtn_vals3[8][1]));
     rts_vals4[8][0] = glass_breakage (&(rts_vals4[8][1]));
    rta_vels5[8][0] = glass_breekage (6(rta_vels5[8][1]));
    rta_vals1[9][0] = effects_comparison (&(rta_vals1[9][1]));
    rtm_vals2[9][0] = effects_comparison (&(rtm_vals2[9][1]));
    rtm_vels3[9][0] = effects_comparison (6(rtm_vels3[9][1]));
    rta_vals4[9][0] - affects_comparison (&(rta_vals4[9][1]));
    rta_vals5[9][0] = affects_comparison (&(rta_vals5[9][1]));
 landuse (rpt, 1)
 FILE *xpt;
 (
    fprintf(rpt, "\fDESCRIPTION OF LAND USE COMPATIBILITY\a\a");
    Sprintf(spt, 'MER: %s MESSION: %s AIRCRAFT: %s MOST

srcid.arr, missiabl.arr, sq mess.arr, month(math[i]);
                                                                  MORTE: 4s/a/a".
    fprintf(spt, "Lend uses competible with the noise exposure pr
  flight\a");
    fprintf(rpt, "operations associated with the proposed action, as specified\
  in the Joint\n");
    fpristf(rpt, "Services Land Use Planning Manual, are as noted below under \
   rst oase\a");
    fprintf(rpt, "assumptions. These land use interpretations are for the MER \
       et and\n");
    fprintf(rpt, "mosth producing the highest moise exposure.\a");
    fprintf(xpt, "\n 0.0 miles from the MTR Centerline: \n\n");
    rtn_vals1[0][0] = habitability (1, ldmar[i][0], &(rtn_vals1[0][1]),rpt);
    fprintf(rpt, "\m 0.5 miles from the MTR Centerline: \m\m");
    rts_wals2[0][0] = habitability (2, ldmar[1][1], &(rts_wals2[0][1]), mpt);

Sprintf(mpt, "\n 1.0 miles from the MFR Conterline: \n\n");
    rtm_vals3[0][0] = habitability (3, ldmar[1][2], 6(rtm_vals3[0][1]), rpt);
    fprintf(rpt, "\m 1.5 miles from the MTR Centerline: \m\m");
    rts_vals4[0][0] = habitability (2, ldmar[i][3], &(rts_vals2[0][1]), rpt);

Sprintf(rpt, "\n 2.0 miles from the MTR Centerline: \n\n");
    rts_vale5[0][0] = habitability (3, ldmar[1][4], &(rts_vale3[0][1]), rpt);
inconsequential (spt, i)
FILE *rpt;
int 1;
   sraid.arr, misslabl.arr, ad name.arr, mosth[math[i]]);
   fprintf(xpt, "
                      The following effects of noise exposure produced by the \
   fprintf(rpt, "operations associated with the proposed action on people, \
structures, or\m");
   fprintf(rpt, "animals were determined to be inconsequential in the current \
 mavironmental\n");
   fprintf(rpt, "assessment:\a\a");
    heck(rpt,0);
mimor(rpt,i, which)
FILE *spt;
   fprintf(upt, "\fdescription of moise effects of minor importance\n \ n");
   fprintf(rpt, "MFR: 4s
                            MISSSION: to AIRCRAFT: to MOSTE: to\n\n^,
           srcid.arr, misslabl.arr, ac_name.arr, month[math[i]]);
   fprintf(rpt. "
                      The following effects of noise exposure produced by the \
   fprintf(rpt, "operations associated with the proposed action on people, \
structures, or\n");
   fprintf(rpt, "animals were determined to be of minor importance in the \
current \n");
   fprintf(rpt, "environmental assessment:\n\n");
    beak (zpt, 1);
```

```
if (which - 1) {
        if (anaoy flag == 1) (
bpl_anaoyanos(rpt);
anaoy_flag = -2;
        if (hearing_flag - 1) {
           bpl_hearing(rpt);
           hearing_flag = -2;
        if (sleep_flag == 1) {
  bpl_sleep(zpt);
  sleep_flag = -2;
        if (livestock_flag == 1) {
           bpl_livestck(xpt);
           livestock flag = -2;
        if (speech flag - 1) {
           bpl_speech(rpt);
           speech flag = -2;
        if (glass_flag - 1) (
           hpl_glass (zpt);
           glass_flag = -2;
       1
   )
 considerable (rpt, 1, which)
 FILE *mpt;
 int 1, which;
   fpristf(rpt, .*\firstruction of noise effects of considerable information <math>a = 1
    fprintf(spt, 'MFR: %s MISSION: %s AINCRAFF %s MONTE: %s/n/n", srcid.arr, misslabl.arr, sc_name.arr, month(math[i]);
                         The following effects of moise exposure produced by the \
    fpristf(rpt, *
 flight\m");
    fprintf(rpt, "operations associated with the proposed action on people, \
  tructures, or\n");
    fprintf(rpt, "animals were determined to be of considerable importance in \
 the current \n");
    fprintf(rpt, "environmental assessment:\a\a");
    check (rpt. 2);
    if (which - 1) {
       if (amoy_flag - 2) {
          bpl_unmoyence(rpt);
           manoy flag = -2;
       if (hearing_flag - 2) {
          bpl_hearing(rpt);
          bearing_flag = -2;
       if (sleep_flag - 2) {
          hpl_sleep(rpt);
sleep_flag = -2;
       if (livestock_flag == 2) {
          bpl_livestak(xpt);
          livestock flag = -2;
       if (speech_flag == 2) {
          bpl_speech (rpt);
          speech_flag = -2;
      if (glass_flag == 2) (
          hpl_glass(zpt);
         glass_flag = -2;
notoconsidered(xpt,1)
FILE *spt;
ist i:
   fprintf(rpt, *\fDESCRIPTION OF REFECTS NOT CONSIDERED IN CORREST \
ENVIRONMENTAL ASSESSMENT\B\B");
   fprintf(rpt, "MER: to MISSSION: to
                                                AIRCRAFT: to MONTE: to\n\n",
           srcid.arr, misslabl.arr, ac name.arr, month[mnth[i]]);
(rpt, " The following potential noise affects were not considered)
   fpristf(rpt, "
  fprintf(rpt, "present analyses: \a\a");
```

```
check(rpt,-1);
     Sprintf (rpt,
                   "\n\nReasons that these potential effects were not considered\
   included(n');
     fprintf(rpt, "insufficient information for evaluation, insufficient \
  precision of \a");
     fprintf(rpt, "estimation of moise emposure, and lack of generally accepted \
      of \a");
     fprintf(rpt, "producing quantitative estimates of magnitudes of potential \
  effects.\z");
  references (rpt)
  FILE *rot:
  1
     fprintf(rpt, "\fReresumces(n\n");
     fprintf(rpt, "Busnel, R. 1978. \"Introduction,\" in J. L. Flotcher and \
  R. G. Busnel (eds), \n");
     fprintf(rpt, "Effects of Moise on Wildlife. Academic Press, New York.\a\a");
     fprintf(rpt, "Cotterees, P. 1972. Somis Boom exposure effects: effects on
   animals.\n");
     fprintf(rpt, "Journal of Sound Vibration 20 (4):531-534.\n\n");
     Sprintf(rpt, "Environmental Protection Agency. 1980. Guidelines for moise \
  impact\n");
    Sprintf(spt, "analysis. Office of Air, Moise, and Radiation, United States \
  Environmental\n");
     Sprintf(xpt, "Protection Agency (VEEPA).\m\m");
     Sprints (rpt, "Environmental Protection Agency. 1974. Information on Levels \
  of\a");
    Sprintf(rpt, "environmental moise requisite to protest public health and \
  welfare with am\m*);
    fprintf(rpt, "adequate margin of safety. EPA 550/9 74 004.\n\m");
     fprintf(rpt, Fletcher, J. L., and Busnel, R. G., eds. 1978. Effects of \
    fprintf(rpt, "on wildlife. Academic Press.\n\n");
    fprintf(rpt, "Hershey, R. L., and Higgins, T. H., eds. 1973. Statistical \
    diction model\n*);
    fprintf(rpt, "for glass breakage from nominal sound booms. Federal \
 Aviation \n");
    fprintf(rpt, "Administration Report FAR RD 73-79.\m\m");
    Sprintf(xpt, "Rinshaw, W. R.; Ball, W. B.; Ladson, T. A.; McMail, E. C. E.; \
 and Taylor, J.\n");
   fprintf(rpt, "P. 1970. An annotated bibliography on animal response to somic
      and\n");
    fprintf(xpt, "other load sounds. Washington, D. C.\n\n");
    Sprints (xpt, "International Civil Aviation Organization (ICAO). 1970. Somio \
 Boom Effects\n");
    Sprintf(rpt, "on the Animal Kingdom. Somio Boom Panel, Montreal, 12 21 \
 October 1970.\n\n*);
    fprintf(rpt, "Heeman, J. and E. P. Besttie. 1985, Avistica Hoise Rffects. \
 FAA(n");
   fprintf(xpt,*ME 05 2. Federal Aviation Administration, Hoise Abstract \
 Branch, \n");
   fprintf(rpt, "Mashington, D. C.\m\m");
    fprintf(rpt, "Hizon, C. W.; Hille, H. K.; Sommer, H. C.; and Guild, H. \
 1968. \"Sonic Booms\n");
   fprintf(rpt, 'resulting from extremely low altitude supersonic flight: \
        meets end\n");
   fprintf(rpt, "observations on houses, livestock and people.\" Aerospace \
Medical Research\n");
   Sprintf(rpt, "Leborstories, Wright-Patterson Air Force Rese, Report MMNL TR \
 58 52.\a\a");
 fprintf(rpt, "Flotkin, Esnneth J.; Sutherland, Louis C.; and Molino, John A.\n");
fprintf(rpt, "1987. \"Environmental Moise Assessment for Military Aircraft\a");
   Sprintf(rpt, "Training Routes. Volume 2: Recommended Noise Matrix.\"
   Asrospace\z");
   Sprintf(rpt, "Medical Research Laboritories, Wright-Patterson Air Force \
   Base, \n");
   fprintf(xpt, "Report AMML-TR-87-001.\m\m");
   Sprintf(rpt, "Shotton, L. R. 1982, \"Response of Wildlife and Farm Animals \
to Low Level(n");
   fprintf(rpt, "Military Jet Overflight.\" The Reporter II (6):161-164.\2");
bpl_livestck(rpt)
FILE *spt;
  fprintf(rpt, "\m\nLIVESTOCK DAMAGE(m\n");
   fprintf(rpt, "The U.S. Environmental Protection Agency (EPA) has reviewed \
the literature\n");
  fprintf(rpt, "on noise effects in domestic animals (Dufour 1980) . In \
general, there is en\n");
```

```
fprintf(rpt, "overall trend for domestic snimels to adapt to intermittent \
     fprintf(rpt, "aircraft-like) noise under 120 dB (decibels). Bussel (1978)\
   reviewed\m*);
     fprintf(rpt, "effects around large airports and found no evidence to \
  indicate noise-\n");
     fprintf(rpt, "related adverse effects.\a\a");
     fprintf(rpt, "Megative behavioral effects from exposure to scale booms are\
   rare among \z );
     fpristf (rpt, "dos
                      estic azimals such as horses, cattle, sheep and poultry \
  (Cottoresm 1972; \m");
     fprintf(rpt, "Fletcher & Busnel 1978; Einshew et al. 1970; Mimon et al. \
  1960: \m");
    fprintf(rpt, "International Civil Aviation Organization [ICAQ] 1970) . Large\
  form saimals\x*);
     fprintf(rpt, may respond with spontaneous activity (i.e. galloping, \
 bellowing, jumping). \n");
    fprintf(rpt, "Poultry show mild reactions to the booms in most cases, but in
  less then teni\n");
    fprintf(rpt, "percent of the cases chickens reacted with growling, gowering, \
  or moise.\m");
    fprintf(rpt, "There was reported to be no significant effect on egg \
 production, milk\n");
    fprintf(xpt, "production, or food consumption. Figs, both in the open and \
 in shelter, \n");
    fprintf(rpt, "show a tendency to be quiet (ICMQ 1970). Observations show \
 greater(2");
    fprintf(rpt, "responses resulting from low-level subsonic flights, \
 motorcycles, paper\n");
    fprintf(rpt, blown by the wind and other startling stimuli (ICMQ 1970). \
 There appears to\n*);
    fprintf(rpt, be no report of panic, injury or negative effects upon \
 reproductive success\n");
    fprintf(rpt, "(Fletcher & Busnel 1978) .\a\a");
 bpl hearing (rpt)
 FILE *spt;
    fprintf(rpt, "\n\nEEARISG DAMAGE RISE\n\n");
    fprintf(rpt, "Rearing loss can be either temporary or parmenent. A \
 moise-induced\n");
    fprintf(rpt, "temporary threshold shift is a temporary loss of hearing \
 experiesced after a\a");
    fprintf(rpt, "relatively short exposure to excessive noise. A noise-induced\
  threshold\n");
   fprintf(rpt, "shift means that the detection level of sound has been \
 increased. Recovery\n");
   fprintf(rpt, "is fairly repid after communities of the noise. A noise-induced\
      exect(x");
    fprintf(rpt, "threshold shift is an irreversible loss of hearing caused by \
prolonged\n");
   fprintf(rpt, "exposure to excessive noise. This loss is essentially \
 indistinguishable\n");
   fpristf(rpt, "from the normal hearing loss associated with aging. Permanent\
    ring loss \n");
   fprintf(rpt, "is generally associated with destruction of the heir cells of \
the inner ear. \n");
   fprintf(rpt, "Based on EPA (Environmental Protection Agency) criteria, \
hearing loss is not\n");
   fprintf(rpt, expected for people living within noise contours below DEL \
levels of 75 dB\n");
   fprintf(rpt,*(decibels). Further, as stated in the EFA \"Levels Dogment,\"\
 changes in\n");
   fprintf(rpt, "hearing levels of <5 dB are generally not considered noticeable\
 OF\R");
   fprintf(rpt, "significant (EPA 1974).\m\m");
bpl_speech(xpt)
FILE *rot:
ſ
  fpristf(rpt, "\a\nsweece INTERGERENCE\a\n");
   fprintf(rpt, *Speech interference associated with aircraft moise is a \
primary source of \n");
  fprintf(rpt, "annoyance to individuals on the ground. The disruption of \
leisure\a");
  fprintf(rpt, *activities (such as listening to the radio, television, music, \
```

```
ead\a") :
    fprintf(rpt, "conversation) gives rise to frustration and irritation. \
 Quality speech\n");
    fprintf (rpt, "od
                       minstion is obviously also important in the classroom, \
 office, and\n");
    fprintf(rpt, "industrial settings. Researchers have found that aircraft \
 moise of 75 dm\m");
    Sprintf(rpt, "(decibels) annoyed the highest percentage of the population \
  when it\n"):
    fprintf(rpt, "interfered with the television sound. Righty percent of the \
 list of\a");
    fprintf(rpt, "annoyances for the surveyed population was flickering of the \
 television\n");
    fpristf(rpt, "picture and interference with casual conversation by mircraft \
 moise (Werman\n");
    fprintf(rpt, "6 Besttie 1985) .\a\a");
 bpl_manoyance(rpt)
 FILE *zpt;
    fpristf(rpt, "\a\naumam Ammorance\a\a");
    fprintf(xpt, "Noise annoyance is defined by the U.S. Environmental \
 Protection Agency (EPA) \n");
    fprintf(rpt, "as any negative subjective reaction to noise on the part of an
  individual or\n");
    fprintf(rpt, "group (EPA 1978). \"Except in the case of speech interference,\
  however, the\n"1:
    fprintf(rpt, "degree of interference is hard to specify and difficult to \
 relate to the \n");
    fprintf(rpt, "level of moise exposure\" (EFA 1978). \"Aircraft moise may \
     . be found(a");
   fprintf(rpt, "annoying because it may startle people, cause houses to shake,\
  or eligit\n=):
   fprintf(rpt, "feer of a crash\" (EPA 1978) .\n\n");
}
bpl_glass(rpt)
FILE *rot:
    fprintf(rpt, "\n\nsrmcronal Damace\n\n");
    fprintf(xpt, "By far, the largest percentage of sonic boom damage claims \
stem from broken \n");
   fprintf(rpt, "or cracked glass. All of the tests conducted in the United \
States have\n");
   fprintf(rpt, confirmed that glass damage is the most prevalent damage caused)
 by scale(n");
   fprintf(rpt, booms (Hershey & Higgins 1973). Because the microstructure of
 glass is\n");
   fprintf(rpt, "amorphous rather than orystalline, the practical design \
strength of glass \n");
   fprintf(rpt, "is depend
                          ent on the surface scratch condition. Glass that has
 been\n");
   fprintf(rpt, "sandblasted, scretched, or micked will not exhibit the same \
strength as \z");
   fprintf(rpt, "a properly installed relatively new pene of glass.\a\a");
bpl_sleep(rpt)
FILE *spt;
   fpristf(rpt, "\s\ssace IFFERFERENCE\s\s");
   fprintf(rpt, "Sleep interference is one of the factors contributing to \
sircraft soise\n");
  fpristf(rpt, "sanoyance. Airport mighttime restrictions have been employed \
to minimise(n");
  fprintf(rpt, "this annoyance. In the case of nighttime operations, as \
   erior maximum\n");
  fprintf(rpt, "sound level (Alm) of 72 dB (decibels) is identified as an \
     table sleep\n");
  fprintf(rpt, "interference condition for a windows-closed condition. This \
corresponds to\n");
  fprintf(rpt, "an interior Alm of about 55 dB.\a\a");
  fprintf(rpt, "To provide a basis for estimation of the number of people who \
could be \n");
  fprimtf(xpt, "awakened by a specific noise event, data developed by \
```

```
Goldstein and Lukes\n");
     fprintf(rpt, *(1980) were used to develop a relationship between the EEL \
  value and the \n");
                       est of exposed persons who would be awakened by the noise \
     fprintf(rpt, "per
  event. These\z"):
     fprintf(rpt, "data indicated that the percent swakened by a specific \
  interior moise level\n");
     fprintf(rpt, "can be expressed by the following equation: \a\a");
     Sprintf(rpt,
                           Percent Amekened = 1.1(ASEL) - 49.5, \n");
    Sprintf(rpt, "where ASEL = the interior A-waighted sound exposure level.\a\n");
    fprintf(rpt, "Since noise must penetrate the home to disturb sleep, interior\
  moise levels\n"):
    fprintf(rpt, "will be lower than outside levels due to the absorption of \
 sound energy\z");
    fprintf(rpt, "attenuation by the structure. The amount of attenuation \
  provided by the \n");
    fprintf(rpt, "building is dependent on the type of construction and whether \
 wiadows are\a");
    fprintf(rpt, "open or closed. The Environmental Protection Agency recommen
  attenuation(a"):
    fprintf(rpt, "factors of 17 dB (decibels) for summartime (windows open) \
 residential\n");
    fprintf(rpt, "conditions and 27 dB for wintertime (windows alosed) \
     ditions.\n=):
    fprintf(rpt, "Incorporating the attenuation factors into the above equation \
 gives the \a");
    fprintf(rpt, "following relationships for the percent awakened under \
      urtime and\n");
    fpristf(rpt, "wintertime conditions:\n\n");
    fprintf(rpt, "
                         Percent Awakened (summer) = 1.1(ASEL - 17) - 49.5\n");
    fprintf(rpt, *
                                                      = 1.1(ASEL) - 60.2(n/n^*);
    fprintf(rpt, "
                          Percent Asskened (winter) = 1.1(ASEL - 27) - 49.5\n");
    fprintf(rpt, *
                                                      = 1.1(ASEL) - 79.2\m\m^);
 check (rpt, which)
 FILE *rpt;
    fprintf(rpt, "\n 0.0 miles from the MRR Conterline: \n\n");
    if (rtn_vals1[1][1] = which) {
    Sprintf(rpt, "Prevalence of Annoyance Among Population\n");
       samoy flag = which;
    if (rts_vals1[2][1] -- which) (
       fprintf(rpt, "Hearing Demage Risk to Residential Population\n");
      hearing_flag = which;
    if (rtn_vals1[3][1] — which) {
      Sleep flag = which;
   if (rtn_vals1[5][1] -- which) {
      Sprintf(xpt, "Reproductive Success or Population Size of an Endangered \
Species \n");
      species flag = which;
   if (rtn_vals1[6][1] - which) {
      fprintf(rpt, "Economic Demage to Livestock\n");
      livestock_flag = which;
   if (rtn_vals1[7][1] - which) {
      fprintf(rpt, "Speech Interference of Residential Population\a");
      speech_flag = which;
   if (rts_vals1[0][1] - which) (
     Sprints(rpt, "Glass Breakage Claims Among Residential Population\n");
glass_flag = which;
  if (rta_vals1[9][1] - which) {
      fprintf(rpt, "Effects Comparison Module\n");
compare_flag = which;
  if (annoy flag != which 66 hearing flag != which 66 sleep_flag != which 66
      livestock flag != which 66 speech flag != which 66 glass_flag != which 66
       compare flag != which && species flag != which)
      fprintf(rpt, "There were no effects of noise in this category at this \
distance\x*);
  fprintf(rpt, "\n 0.5 miles from the MTR Centerline: \n\n");
  if (rta_vals2[1][1] - which) {
```

```
fprintf(rpt, "Frevalence of Annoyance Among Population\n");
         annoy flag = which:
     // (rtn_vals2[2][1] -- which) {
    Sprintf(rpt, "Esaring Demage Risk to Residential Population\n");
    hearing_flag = which;
     if (rtm_vals2[3][1] - which) {
         fprintf(rpt, "Sleep Interference of Residential Population\n");
sleep flag = which;
     ;
if (rtn_vels1[5][1] - which) {
    Sprintf(rpt, "Reproductive Success or Population Size of an Endangered \
     ("E/bolbs
         species flag - which:
     if (stn_vals2[6][1] — which) {
         fprintf(rpt, "Economic Demage to Livestock\n");
         livestock flag = which;
     if (rta_vals2[7][1] - which) {
        printf(rpt, "Speech Interference of Residential Population\n");
speech flag = which;
     if (rts_vals2[0][1] - which) {
        Sprintf(rpt, "Glass Breakage Claims Among Residential Population\n");
glass_flag = which;
     if (rts_vals2[9][1] - which) (
        fprintf(rpt, "Effects Comparison Module\x");
compare flag = which;
     if (annoy_flag != which 66 hearing_flag != which 66 sleep_flag != which 66
         livestock flag |= which 66 speech flag |= which 66 glass flag |= which 66 compare_flag |= which 66 species_flag |= which)

Sprintf(rpt, "There were no effects of noise in this outegory at this \
 distance\n");
     fprintf(rpt, "\n 1.0 miles from the MFR Centerline: \n\n");
    if (rtn_valss[1][1] = which) {
    fprintf(rpt, "Frevalence of Annoyance Among Population\n");
    annoy_fing = which;
    if (rts_vals3[2][1] - which) {
       fprintf(rpt, "Bearing Demage Risk to Residential Population\a");
hearing_flag = which;
    ("E/seto
       species_flag = which;
    if (rtn_vals3[6][1] -- which) {
       fprintf(rpt, "Economic Damage to Livestock\n");
livestock_flag = which;
   if (sta_vals3[7][1] -- which) {
       fprintf(rpt, "Speech Interference of Residential Population\n");
       speech_flag = which;
   if (rts_vals2[8][1] == which) {
       fprintf(spt, "Glass Breakage Claims Jeong Residential Population\n");
glass_flag = which;
   ;
if (rts_vals3[9][1] — which) {
    fprintf(rpt, "Effects Comparison Module\n");
    compare_flag = which;
   if (amony flag != which 66 hearing flag != which 66 sleep flag != which 66
        livestock flag != which 66 speech flag != which 66 glass flag != which 66
       compare_flag != which 66 species flag != which)

fprintf(rpt, There were no effects of noise in this category at this \
distance\n");
   fprintf(rpt, "\n 1.5 miles from the MTR Centerline: \n\n");
   if (rtn_vals4[1][1] = which) {
   fprintf(rpt, "Frevalence of Annoyance Among Population\n");
   eaboy_flag = which;
```

```
if (rta_vels4[2][1] - which) (
          Sprints(spt, "Searing Dumage Risk to Residential Population\n");
hearing flag = which;
      if (rta_vals4[3][1] - which) {
          fprintf(rpt, "Sleep Interference of Residential Population\n");
sleep_flag = which;
       /
if (rta_vals4[5][1] - which) {
          Sprintf(spt, "Reproductive Success or Population Size of an Endangered \
       : ("g/solp
         species_flag = which;
      if (sta_vals4[6][1] - which) {
         Sprintf(rpt, "Economic Demage to Livestock\a");
          livestock flag = which;
      if (sta_vals4[7][1] - which) (
         fprintf(rpt, "Speech Interference of Residential Population\n");
speech_flag = which;
      if (rtm_vals4[8][1] - which) {
         Sprintf(rpt, "Glass Breakage Claims Among Residential Population\a");
glass flag = which;
      1f (rtn_vels4[9][1] - which) {
         fprintf(rpt, "Effects Comparison Module\n");
compare_flag = which;
     if (assoy_flag != which 66 hearing_flag != which 66 sleep_flag != which 66
          livestock flag != which 66 speech flag != which 66 glass flag != which 66
          compare flag to which && species flag to which)
          Sprintf(rpt, "There were no effects of noise in this category at this \
  distance\n");
     fpriatf(xpt, "\a
                         2.0 miles from the MTR Centerline: \m\m");
     if (sta_vale5[1][1] == which) (
        fprintf(rpt, 'Frevelance of Annoyance Among Population\n');
annoy_flag = which;
     if (rts_vals5[2][1] - which) {
        fprintf(rpt, "Searing Demage Risk to Residential Population\a");
        hearing_flag = which;
     if (sts_vals5[3][1] - which) {
        fprintf(spt, "Sleep Interference of Residential Population\a");
sleep_flag = which;
     if (rts_vals5[5][1] - which) {
        Sprintf(xpt, "Reproductive Success or Population Size of an Endangered \
 Species\n*);
        species_flag = which;
    if (rts_vale5[6][1] - which) (
       fprintf(rpt, "Economic Demage to Livestock\n");
livestock flag = which;
    if (rta_vals5[7][1] - which) (
       Sprintf(xpt, "Speech Interference of Residential Population\n");
       speech_flag = which;
    if (rts_vals5[8][1] - which) {
       Sprints (spt, "Glass Breakage Claims Jesong Residential Population\n");
glass_flag = which;
   if (sta_vals5[9][1] == which) {
      fprintf(rpt, "Effects Comparison Modele\n");
compare_flag = which;
   if (annoy_flag != which && hearing_flag != which && sleep_flag != which &&
       (analy fing |= which 55 hearing fing |= which 55 sleep fing |= which 55 livestock fing |= which 55 speech fing |= which 56 glass fing |= which 56 compere_fing |= which 56 speecies_fing |= which)

fprintf(rpt, "There were no effects of noise in this category at this \
distance\n");
rptasen.pd -- ASAN Main Program (Temporary driver for the
                           report generator
```

```
finalude <stdio.b>
                                      /* The usual stuff, of course */
                                      /* Header for calls to MM-Dos */
/* String manipulation header */
  disclose ctime.k
  MANC SQL BRGIN DECLARS SECTION:
  EXEC SQL INCLUDE hostwars.k;
  MING SQL INCLUDE faharris.h;
  EXEC SOL END DECLARS SECTION.
  EXEC SQL INCLUDE SQLCA;
                                      /* SQL Communication Area
                                                                         */
  #define SQLCA_STORAGE_CLASS
  finolude "asen.h"
                                      /* Standard ASAN Meader file */
                                      /* In agan.pc it must be here */
/* since we need to initialize */
                                      /* character arrays!
  Char targy[15];
 main (argo, argv)
  int ergo;
cher *ergv[];
  int i, j, myonama();
  static char *legal_motice[11] =
  ("\B\B\B\B\t\t\tRESTRICTED RIGHTS LEGED\B\B",
  '\tUse, duplication, or disclosure is subject to restrictions\a",
 "\t as set forth in subdivision (b) (s) (la) va week."

"\t Rights in Technical Data and Computer Software Clause\n",
 "\t\t\tEN LABORATORIES INCORPORATED\".
          10 MOULTON STREET\n",
 "\t\t\t
 "\t\t\t
             Combridge, MA 02238\n*,
 */$/$/$
                617-873-3000\a\a\a",
       User Interface Copyright (C) 1985, RMM Laboratories Encorporated\n",
 "\t\t\t
             All Rights Reserved");
 exters cher *melloc();
 wold emponing();
 FILE *fopen();
    printf("\nUsage: RFTAINN assessment name printfile name");
printf("\n The printfile name is optional: ");
printf("if unspecified printer [prn] is assumed.\n\n");
    erit (16);
 printf("40[25", 27);
 for (i = 0; i < 11; i++) printf("%s",lagal_notice[i]);
printf("\m\m\n\n\t Please tap the space ber to continue, \"CTRL-C\" \
 to abort. \a\a");
 do {
     1 = getah();
     } while (i != ' ');
printf("ASAW Report Module starting....");
stropy(workspace.arr, argv[1]);
strust (workspace.arr, "/");
street (workspace.arr, univowd);
workspace.len = strlen(workspace.arr);
EXEC SQL COSSECT : workspace ;
if (sqlos.sqloods) ( EEECass);
      printf("\mlogon failed\mts", sqlon.sqlerms.sqlerms); exit(4);}
EXEC SQL SELECT myld, myprivs, mysens
                FROM Sys . Tempeser
                DETO :userno, :O auth, :my name;
my_name.arr[my_name.len] = '\0';
stropy (Assessment.name, my_name.arr);
                                         /* Store results in global */
/* ASSESMENT structure */
ASSESSMENT.id - mearno;
EXEC SQL SELECT description FROM table of contents INTO :workspace
         WEERE idnumber = :userno;
if (sqloa.sqloods) {
  printf("\mlogon: ts", sqlon.sqlerms.sqlerms);}
```

```
alse {
    j = (sixeof ASSESSMENT.desc) -1;
    if (workspace.las >= j) {
        for(i = 0; i < j; i+t) ASSESSMENT.desc[i] = workspace.arr[i];
        ASSESSMENT.desc[j] = '\0'; }
    else    stropy(ASSESSMENT.desc, workspace.arr);
}

if (argc > 2)
    stropy(targv, argv[2]);
    else
        stropy(targv, "pra");

Uinit();
    printf("bc[24;1E",27);

exit(2);
}

callrpt() {
    report(targv);
}

callrpt3() {
    report3(targv,0);
}
```

## **B.1** C Language Source Code

```
ASAN Referets Modules
     Created: January 8, 1987
    The return or success codes returned by the procedures in this
     module have the following meanings:
/* /* /*
           -2 = execution is not possible because the module is
              not implemented yet
           -1 = execution is not possible because the available
              imput is incomplete or improper
           0 = the precision of the estimate is satisfactory
           1 = the module executed, but the precision of the
               estimate is unsatisfactory
    The severity of effect codes returned by the procedures in this
    module have the following meanings:
          -1 = effect not considered in the current analysis
           0 = magnitude of predicted effect is inconsequential
           1 = magnitude of predicted effect is of minor importan
/*
           2 = magnitude of predicted effect is of considerable
              importance
finciade "stdio.h"
1, 0, 30, 25, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1,
                   0, 0, 0, 1, 1, 1, 1, 1, 1, 1);
char table2[20][80] = { "Family housing", "Bachelor housing",
                   "Transient lodging - hotels, motels",
                   "Classrooms, libraries, churches",
                   "Offices and administration buildings, military",
                   "Offices - business and professional",
```

```
"Hospitals, medical facilities, nursing homes",
                      "Dental climic, medical dispensaries",
                      "Outdoor music shells",
                      "Retail stores, restaurants, banks, movie theaters",
                     "Flight line operations, maintenance and training", 
"Industrial, manufacturing and laboritories",
                     "Outdoor sports areas, outdoor spectator sports",
                     "Playgrounds, active sport recreational areas", 
"Meighborhood parks", "Gymnasiums, indoor pools",
                     "Outdoor - frequent speech communication",
                     "Outdoor - unfrequent speech communication",
                     "Livestock farming, sainel breeding",
                     "Agricultural (amospt livestock)" );
habitability (which, ldm, effectoods, rpt)
double lda;
int
         *effectoode, which;
         *zpt;
/* The above tables and below look-up computations were derived from
   the Joint Services Hoise Planning Manual as referenced in Table 6.3 of Theodore J. Schultz's book, Community Hoise Esting, Second Edition
    (1902) .
ŧ
  int i, index;
  cher zeme[15];
  if (lda > 65 && lda <= 69) {
     index = 4;
      *effectoode # 1:
  also if (lds <= 74) {
     index = 3;
     *effectoods = 2;
  also if (1da <- 79) {
     index = 2:
     *effectoode = 2;
  else if (lds <= 84) {
     *effectoode = 2:
  else if (lds cos) {
     index = 0;
     *effectoode = 2;
 fprintf(rpt, *
                        *s \n", table2[1]);
     *effectoode = 0;
 *effectoode = 2;
 alse (
    fprintf(xpt, " The following are competible land uses: \n");
    Sprints (spt,
       "\n The following land uses are compatible with interior moise\n");
    fprintf(rpt," reduction noted in parentheses:\n");
for (1 = 0; 1 < 20; 1++)
       if (table[i][index] != 1 && table[i][index] != 0)
          fprintf(rpt,
                            ## (%d) \n", table2[i], table[i][index]);
   sprints(spt, "\n The following are incompetible land uses:
for (i = 0; 1 < 20; i++)</pre>
      if (table[i][index] -- 0)
          fprintf(rpt, "
                           % \x",table2[1]);
 return (0) ;
```

```
anacymnos (ldn, effectedds)
 int
         *effectoode:
  /* The below calculation of percent of the population that will be highly
      sanoyed by a sound of level Ldn was copied from Theodore J. Schultz's article, "Synthesis of social surveys on noise annoyance", published in the Journal of the Accestical Society of America, Vol.64, No. 2, Amgust
      1978, pp 377-405.
     double perha, pow();
     1f (ldm < 40 || ldm > 85) (
         *effectoods = -1:
        return (-1);
           rha = 0.8553 * 1da - 0.0401 * pow(lda,2.) + 0.00047 * pow(lda,3.);
         if (perbs < 5)
            *effectoode = 0;
        else if (perts <= 10)
            *effectoode = 1:
        else if (perha <= 20)
            *effectoode = 2;
        معله
            *effectoods = 2;
       return (0) ;
 hearing_damage (alevel, effectoods)
 double alevel;
          *effectoode;
 /* The calculation of ea was derived from William Burns' book, Hoise
     and Man, Second Edition (1973), page 238. The dB loss values were
      read from the 50% curve on the bottom graph on page 239 of the same
    int es, temp;
                              /* en = A-weighted noise immission level
    es = (int) slevel + 10; /* 10 is derived from 10*log(duration of sound
                                     emposure in years) which is assumed to be 5
                                     years here
    temp = ea 4 5;
                              /* round to the meanest 5 dbs
    1f (temp <= 2.5)
        es - temp;
        ea += (5 - temp);
/* en of 85 -> 50% of the population would experience a 1.2dB hearing loss */
/* 90 -> 2.3dB loss, 95 -> 4.5dB loss, 100 -> 8.7dB loss */
/* 105 -> 14.1dB loss, 110 -> 22.3dB loss, 115 -> 31.2dB loss */
/* 120 -> 40.0dB loss, 125 -> 43.1dB loss, 130 -> 43.1dB loss */
   if (ma <= 95)
       *effectode = 0:
   else if (ee <= 110)
       *effectoode = 1;
   وعله
       *effectoode = 2;
   return (0) ;
sleep_interference (alevel, effectcode,i)
double slevel;
         *effectoods, 1;
  double per_ewekened;
   if (i > 4 && i < 10) /* stacks */
per_awekened = 1.1 * alevel - 68.2;
                              /* WINTER */
      per_awakened = 1.1 * alevel = 79.2;
   if (per_swekened <= 5.0)
```

```
*affectoode = 0;
also if (per_awakened <= 10.0)
*affectoode = 1;
       *effectoode = 2;
    return (0) ;
 sem_land_use (effectuade)
int *effectuade;
    *effectcode = -1;
    metuma (-2);
 endangered_species_reproduction (affectsode)
int *effectsode;
   *effectcode = -1;
   return(-2);
 livestock_damp (effectsode)
int *effectsode;
    *effectoode = -1;
   return (-2);
 speech_interference (effectsode)
int *effectsode;
   *effectoode = -1;
   return (-2);
 glass_breakage (effectoode)
int *effectoode;
*effectoods = -1;
  return (-2);
effects_comparison (effectoods) ist *effectoods;
   *effectoode = -1;
CR.C -- ASAN CRASS color willities.
          THIS CODE ASSUMES 256-COLOR LUT AND 2 OVERLAY PLANES.
 finciade "grass.h"
finciade "gglobs.h"
imit_colors ()
 - dark 3;
   /* 777 ( C_CURSOR , 255 , 255 , 285 ) */
   InitCI ( C_SCR_FRAME, 255 , 255 , 255 );
```

```
PaitCI( C_WIM_FRAME, 255 , 255 , 255 );
PaitCI( C_LMG_FRAME, 255 , 255 , 255 );
      TaitCI( C LEG TEXT , 255 , 255 , 255 );
                         , 1, 1, 1);
, 255, 0, 0);
, 0, 255, 0);
, 0, 0, 255);
      Initci ( C_MACK
      Dalter ( C RED
      InitCI ( C COCKE
      InitCI ( C BLUE
      Pater( C_MELOW , 255 , 255 , 0 );
Pater( C_MELOW , 255 , 0 , 255 );
Pater( C_CYAN , 0 , 255 , 255 );
      /* for DMA elevations, use a smooth palette at low intensity */
      ig_palet( C_1ST_DMA , NOM_DMA_C , 0.25 , 0.50 );
      /* for estegories, use a repesting smooth pelette at high intensity */
      for ( j = C_1ST_CAF ; j <= MAX_C - C_CAT_REPRAY + 1 ; j += C_CAT_REPRAY )
          ig_palet( ) , C_CAT_REPRAT , 0.50 , 0.50 );
      42 ( ) < MAX 0)
          ig_pelet( j , max_c - j + 1 , 0.50 , 0.50 );
     maxt_cet_pv = C_1ST_CAT; /* initialize pizvel to use for cetegory */
 ig_pelet ( fpv , malrs , i , s )
                                        /* load smooth palette in Imagraph LUT */
                                            /* first LOT location to load */
   int
         aglre:
                                            /* how many LOT locations to load */
   domble 1, s;.
                                            /* intensity, saturation */
     double hims, h, r, g, h;
     if ( (fpv < 0 ) || ( aclrs < 1 ) || (fpv + aclrs - 1 > MAX_C ) )
         SLOUTS ("7ig_palet: bad parameter");
         return ( -1 );
     hims = 360.0 / (double) malre;
     for ( pv = fpv ; pv < fpv + malrs ; pv++ )
         hisrgb( h , i , s , &r , &g , &b );
         TaitCI( pv, dround(r * 255.0), dround(g * 255.0), dround(b * 255.0) );
         h += hina;
 * hisrgb() -- converts (h,i,s) color to (r,g,b) intensities
               hiergb( h , i , s , &r , &g , &b );
 * calling args (normalized before use):
     0.0 <= h <=360.0 (red = 120.0, green = 240.0, blue = 0.0)
0.0 <= i <= 1.0 0.0 <= s <= 1.0
    0.0 - 1 - 1.0
   returned args:
     0.0 <= z <= 1.0
                        0.0 <= g <= 1.0
                                             0.0 🗢 b 🗢 1.0
histor (h,i,s,r,g,b)
  double h, i, s, sz, sg, sb;
   /* mormalize calling args */
    While ( 2 < 0.0 )
                        h += 360.0;
    While ( h > 360.0 ) h -= 360.0;
   While (1 < 0.0) 1 += 1.0;
While (1 > 1.0) 1 -= 1.0;
   while ( # < 0.0 )
   while ( # > 1.0 )
   if (i > 0.5) n = i + s - (i * s);
also n = i * (1.0 + s);
```

```
m2 = ( 2.0 * 1 ) - m;
       if \{h < 60.0\} %r = m2 + (m - m2) * (h / 60.0);
else if (h < 180.0) %r = m;
else if (h < 240.0) %r = m2 + (m - m2) * ( (240.0 - h ) / 60.0);
      if \{h < 120.0\} q_0 = m2; else if \{h < 180.0\} q_0 = m2 + \{m - m2\} f(h - 120.0) f(h < 180.0) q_0 = m2; else if \{h < 180.0\} q_0 = m2
                               ^{4}g = m2 + (m - m2) + ( (360.0 - h) / 60.0 );
      if \{h < 60.0\} %b = m;
else if \{h < 120.0\} %b = m2 + \{m - m2\} * \{\{120.0 - h\} / 60.0\};
else if \{h < 240.0\} %b = m2;
      also if ( h < 300.0 ) % = m2 + ( n - m2 ) * ( ( h - 240.0 ) / 60.0 );
      alse
                                                                /* questionable */
  /* end of on.a */
  desan.c -- dummy asem mainline to test map screen features
   #define DB (str)
   * needs datum for where's my finger on map.
     show coordinates of specified location on map.
 #include <stdio.b>
 #include "asantype.h"
#include "asan.h"
                   /* DEFINED IN GRAPHICS CODE TO BE SUPPLIED BY FME: */
 double shudist;
 extern cher dlacm[][12];
                  /* ASSUMED TO BE DEFINED ELSEWHERE IN ASAH THER CODE: */
 COORDINATE est;
 COCRDINATE show;
 ASSMERADE ASSESSED
 cher idptofint[6];
 int shwaresing;
 int shwareslow1;
 int shwaresPSF;
int shwareePSF1;
extern int MEA DEBUG FEATURES: /* for debug */
main ()
    Ginit ();
                                   /* imitialize graphics */
    DB( stropy( dlnam[0] , "FirstMap" ); )
DB( stropy( dlnam[7] , "JustaMap" ); )
DB( stropy( dlnam[14] , "Last_Map" ); )
    ENA_DEROG_FRATORES = 1;
                                 /* emable T debugging features */
    Otalt ();
                                   /* disappear into U */
   SLOURS (**** inside dummy() ****);
latides ( est )
 int ent;
   SLOUIS ("*** inside Int2dec() ***");
```

Ð

```
log2des ( est )
   int ent;
   ł
     $LOUTS ("*** inside lon2des() ***");
 ASAMoonn ( arr )
   unsigned ther *exr;
     SLOTER ("*** inside ASAHooma() ***");
 /* end of desea.c */
 do.c -- ASAW CRASS call file handlers.
  #define FUTRASTON 0
                             /* = 1 if okey to use PutRast */
 #define DB (str)
 ∯include "grass.h"
∮include "gglobs.h"
 #include <fort1.b>
 #include <sys\types.b>
 finclude <sys\stat.b>
 finciade (io.b)
 extern char MCA_elev_being_drawn;
 static cher
               fam[80];
               £;
 static ist
 static int
               abytes;
static int
               SCETOW;
static char
               mbuf [MANOOLS];
static cher
               what [MANDOLE];
d_coll ()
   int r, e;
   int od, daw; /* drawing-method-test only */
    set_display_peress ();
                                    /* set up window & viewport */
   sprintf( fnem, "%s\\%s\\asll\\%s",
     gisdbase, mapset, location, layer2add);
   if ( (fh = open(fnem , o_NDONLY | o_BINGRY ) ) == -1 )
       SLOUZE ( "?d_call: open to failed" , face );
       return( -1 );
   SCITOW - 0;
# 1f PUTRAFFOR
                     /* color is wired in cell file */
# also
              /* major kludge elert */
# endif /* PUTRAFIOE */
   clip_to_window();
   for ( r = 0 ; r < W_mrows ; r++ )
       if ( ( abytes = reed( fk , mbuf , W_mools ) ) := W_mools )
          DCERR( "to out of data at row td" , famm , r );
          return( -1 );
     If PUTRASTOR
      if ( TMG_elev_being_drawn )
                                          /* fiddle with colors */
          for ( on = 0 ; on < W nools ; on++ )
           if ( rbuf[ca] != 0 )
              rbuf[oc] = ( (int) ( ( (double) rbuf[oc] / 17. ) + 1. ) );
```

```
FutRast( \underline{w}1 , \underline{w}t - sorrow , \underline{w}1 + \underline{w} mools , xbuf );
          for { od = 0 ; od < W_acols ; oc++ )

if ( mbsf[od] t= 0 )
                  SetFG( mbuf[cc] );
                  aLiae( W_1 + oc , W t - serrow ,
    W_1 + oc + 1 , W t - serrow + 1 , 0x40 );
        endif /* PUTRAFIOE */
 # Afdet DOWN_CARE
     if ( ( mbytes = read( fh , mbuf , W_mools ) ) > 0 )
         DCERR( "4s has data left over (4d bytes)" , fasm , abytes );
         return ( -1 );
 # endis /* DONT_CARE */
     alose (fh );
     alip_to_screen();
     return(0);
 serees2cell ()
     ELCOTE ( "screenfoell() is not yet available" );
 static DCERR (a, b, c, d, e, f, g)
    char megbuf[80];
char tmpbuf[80];
    oprintf( tmpbuf , a , b , a , d , a , f , g );
sprintf( megbuf , "7d_cell: " );
stroat( megbuf , tmpbuf );
    SLOUTS ( magbal );
     close( £2 );
    alip_to_surees();
/* and of da.a */
dd.c -- AEAN GRAES code to draw a "digit" file on the screen.
 ****************************
#define DB (str)
#include "grass.h"
finalude "gglobs.h"
finclude (font1.b)
#indlade <sys\types.h>
#indlade <sys\stat.h>
fincindo cio.b>
static char
                Znam [80] ;
static FILE
                *dfp;
static char
                mbuf[100];
static ist
                £line;
static char
               b_type;
                                                        /* block type */
static int
d_digit ()
```

```
int i, j, z, y;
       int last_sig; /* index to latest non-whitespace char (-1-mone) */
       set_display_parens ();
                                             /* set up window & viewport */
       sprintf( fam, "%s\\%s\\digit\\%s",
  gischese, mapset, location, layer2add);
       if ( ( dfp = fopen( fam , "z" ) ) == BULL )
           SLOUTS ( "?d_digit: fopen to failed" , famm );
           meturn ( -1 );
       fline = 0;
       /* skip the 14 lines of unwested header stuff */
       for ( 1 = 0 ; 1 < 14 ; 1++ )
           fline++;
           if ( fgets ( mbuf , 100 , dfp ) - MOLL )
              DDERR( "fgets error" );
               return ( -1 );
      clip_to_window();
       * **** ELUDGE ALERT:
       * We know that current digit files contain amoutly one outagory each,
       * and that next_cat_pv has just been incremented in build legend(), so the following color selection code is MCCE simpler than it ought to be.
       * Color/legend stuff will be VERY screwed up if may of the subsequent
          error returns is taken.
  - SetFG( next_ost_pv - 1 );
     /* process the digit file one block at a time */
     while ( ( b_type = fgets( dfp ) ) i= mor )
 DB (SLOTTS ("dd: 1st char of block = &c = &c",b_type,b_type);)
         switch( b_type )
              0000 'A':
                                                           /* polyline */
                      1f ( fgets( dfp ) 1= ':' )
                         DDERR( "bad char follows block type" );
                         return( -1 );
 DB (SLOTER ("dd A: got ':'");)
                     if ( fscanf( dfp , "%d" , &count ) < 1 )
                         DDERG( "bad count or early mor" );
                         return( -1 );
DB (SLOUTS ("dd A: got count=4d", count);)
                     fline++;
                     if ( focuse( dfp , * %lf %lf * , &dxl , &dyl ) < 2 )
                         DDERR( "bed 1st wester or early BOF" );
                         return ( -1 );
•
                         fline++;
                        if ( fecanf( dfp , * tlf tlf *, tdx2, tdy2) < 2 )
                            DDERR( "bad data or early BOF" );
                             return( -1 );
                        }
DB(printf("dd A: got next vertex elf elf, call d_line()\n",dx2,dy2);)
                        d line();
                        dz1 = du2;
```

```
dy1 = dy2;
  DB (SLOUER ("dd A: dome");)
                0846 'T':
                                                                  /* test */
                        fline++:
                         1f ( fgeta( dfp ) != ':' )
                             DDERR( "bed char follows block type" );
                             return( -1 );
 DB (SLOUZE ("dd T: got':'");)
                        if ( fgets ( mbef , 100 , dfp ) == MULL )
                             DDERR( "bed string or early BOF" );
                             return ( -1 );
                         alse
                         ŧ
                             i = 0; j = 0; last_sig = -1;
                             while ( shuf[i] := '\setminus n' )
                                 if ( ( mbuf[i] = ' ' ) || ( mbuf[i] == '\t' ) )
if ( last_sig == -1 ) /* toes leading white */
                                                                /* keep embedded white */
                                          tst[j++] = zbuf[1++];
                                 else if ( mbuf[i] = '\0' ) /* embedded mull */
                                     break;
                                                                 /* regular old char */
                                     last_sig = j;
txt[j++] = xbuf[i++];
                            txt[last_sig+1] = '\0';
                                                          /* toss trailing white */
DB(SLOUTS("dd I: got '%s'",txt);)
                       fline++;
                        if ( facesaf( dfp , " dif dif " , doxi , ddyi ) < 2 )
                            DDERR( "bed location or early BOF" );
                            return ( -1 );
DB(printf("dd T: got loc %lf %lf\a",dxl,dyl);)
                       flize++;
                       if (fscanf( dfp, " bif bif bif bif bd bc ", sheight, Swidth, Srotation, Selant, Sfont, Sjust) < 6 }
                           DDERR( "bed deta or early BOF" );
                            return( -1 );
DB (SLOOTS ("dd T: got parameters, call d_text()");)
                      d_text( UMA COCONDS );
DB (SLOUTS ("dd A: done");)
                      break.
                      fline++;
                      DDERR( "had block type" );
                      return( -1 );
        }
    folose( dfp );
    alip_to_screen();
    return(0);
static DDERG (a, b, c, d, e, f, g)
    char megbuf[80];
    char tmpbef[80];
   sprintf( tmpbuf , a , b , a , d , a , f , g );
sprintf( megbuf , "?do %s (%d): " , fnem , fline );
stroat( megbuf , tmpbuf );
   SLOOMS ( magbaf );
falose ( dfp );
```

```
alip_to_sarees();
}
  /* end of dd.a */
  draw.c -- code to draw actual graphics
   Sdefine DB (str)
  finclude "grass.h"
finclude "gglobs.h"
  #define XPIX(xx)
                       ( W_1 + drownd( ( xx - V_west ) / UNM per_pixel ) )
( W_b + drownd( ( yy - V_south ) / UNM per_pixel ) )
  #define TPIX(XX)
  extern int
                Efcet, yfort;
                                     /* Imagraph test params */
  extern int
                chardepth, charwidth;
  static int
                t_size = 1;
                                    /* text soom factor (1-16) */
  statio ist
                mpir, ypir;
  () معنلیه
                              /* draw a line */
   (
aLine( MPIX( dzi ) , MPIX( dyi ) , MPIX( dzi ) , MPIX( dyi ) , 0x40 );
 d_text ( otype )
                              /* draw text */
   char stype;
                              /* SCREEN_COORDS or USE_COORDS */
     int i, mf, yf;
     if ( stype - van_coomes )
        क्ष = अध्य ( dal );
        ypix = TPIX( dyl );
     also /* SCREEK_000006 */
        zpiz = izl;
        ypis = iyi;
    t_sise = 1;
    1f ( fost - 'R' )
    xf = xpix - ( strlen( txt ) * charwioth );
also if ( fout == 'C' )
       xf = xpix + ( ( strlen(txt ) * cherwidth ) / 2 );
* /* assume foot == 'L' */
    alse
       mf = mpin;
    Yf = Ypix - ( chardspth / 2 ); /* supplied Y is vartical contex */
    for ( i = 0 ; i < strlea( txt ) ; i++ )
       most = mf:
       yfost = yf;
DB(SLOOTS("Spata: zfost=4d yfost=4d char=4c=0to", zfost, yfost, tzt[i], tzt[i]);)
       /* ignore otrl chars altogether */
       also if ( txt[i] < 040 )
                                   /* send everything also to sureen */
       {
          gputa( tst[i] , t_size , 0 );
xf += oherwidth;
   }
   SetPeta ( -0 );
                                   /* restore solid pattern */
/* end of draw.c */
```

```
gglobs.s -- globals and parameters for ASAE GRASS.
           seess Warring: This Fill Most March gglobe.h *****
   finclude "grass.h"
         dinom[MAX_DLAYERS] [12]; /* displayed layer names */
  Char
  int
         awa dlavers:
                                /* how many layers now displayed */
 LLTER
         leg[MAX CATS];
                                /* legend/out data structure */
  int
         num_llines;
                                /* how many legend/out data entries now */
 4-4
         mast_ost_pv;
                                /* mest pisval to use for ostegory */
                                /* Gurrent mapped ("ASAH") */
/* Gurrent location name ("Sells") */
 حمض
         *mapset;
 cher
         *location:
  Char
         *gledbase;
                                /* Gurrent map data base dir ("\grass\maps") */
  char
         vposmo[12];
                                /* GEFFERT Viewport name ("C"-scerse, etc.) */
         winnene [12] ;
                                /* GEFFERT Window name */
         layer2edd[12];
                                /* name of layer to add to display */
  حمخه
         layer2del[12];
                                /* name of layer to remove from display */
                                /* name to use for layer being saved */
 char
         layersaveneme [12];
 char
         laytype;
                                /* layer is: 'C'= cell, 'D'= digit, 'L'= DLG */
 Char
         window_frame_drawn;
                                /* YES if window frame drawn, also NO */
                                /* YES if legend drawn, else NO */
/* YES if DMA elevations are drawn, else NO */
 char
         legend drawn;
         DER elev drawn;
 double dz1, dz2, dy1, dy2;
                                /* some vertices */
 int
         iz1, iz2, iy1, iy2;
                                /* some vertices */
         tat [100];
 char
                                /* data meeded to draw text */
 double
        height, width, rotation, slant;
 int
         foot;
         just;
 int
        W_t = SCR_T;
                                /* GEFFERT Window, in screen coords */
        Wb = SCR_B;
W1 = SCR_L;
 124
 ist
        TE - SCA R
 int
         W Brows;
                                /* how many rows */
 int
                               /* how many columns */
        W_moole;
 ist
        V_proj;
                               /* current viewport, in UTMs */
 ist
        V_E080;
 double V_morth;
        V south;
 double V_west;
double V_east;
 double V mg res;
 double V_ew_res;
 1st
        V formet:
double Uniper_pizel:
/* end of gglobs.c */
grass.c -- ASAN graphics main routines
 ******************************
#define DB (str)
#include "gress.h"
finalude "gglobs.h"
Char
               DMA elev being drawn;
static char dfit_mapset[]
               = { 'a','a','a','a','a','\o' };
static char dflt_location[] =
               { '#','*','1','1','#','\or };
static char dfit_gisdbase[] =
              { '\\','g','z','ar,'er,'e','\\','m','ar,'p','s','\or.};
static cher dflt_winneme[] =
               {'w','1','m','d','o','w','\o');
static cher
               fnem[80];
static PILE
               *£p;
static int
               after Utait;
static char
               del_layer_im_progress;
                                             /* KLUDGE ALERT */
```

```
Ginit ()
                                        /* call me first */
       mapset = (char *) dflt_mapset;
                                                /* used to be environment wars ... */
       location = (char *) dflt location;
gisdbase = (char *) dflt gisdbase;
stropy( winners , dflt winners );
      hegia_IG_graphics();
          load_default_pelette(); **/
      InitFont();
       init_colors();
       after_Visit = NO;
       clear_screen();
       after Visit - TES;
      DMA_elev_being_drawn = 20;
  mow_wiew ( watz )
                                       /* change viewport */
    Char Posts;
      if ( stromp( vetr , vpasse ) om 0 )
                                                       /* ignore if no change */
      if ( stromp( vetr , "Sells.C" ) == 0 ) ||
  ( stromp( vetr , "Ajo.N" ) == 0 ) ||
  ( stromp( vetr , "Sells.N" ) == 0 ) ||
  ( stromp( vetr , "Ajo.F" ) == 0 ) ||
  ( stromp( vetr , "Sells.F" ) == 0 ) )
           stropy( vpasse , vetr );
           HEWVALS ();
           clear_screen();
      معته
          SLOUZE ( "? maknowa view ts" , vetr );
 clear_screes ()
                            /* erase screen & clear parameters */
     1mt 1:
     erase_graphics_display();
     for ( i = 0 ; i < MAX_DLAYERS ; i++ ) /* no layers displayed */
         stropy( dlam[1] , "" );
     aum_dlayers = 0;
     if ( after_Unit )
                                     /* MENVALS before Uinit -> bizarre creshes */
          HEWVALE ();
erase_graphics_display ()
                                    /* just erase the screen */
     alip_to_screen();
     els ( C 2005 );
                                      /* clear screen to background color */
     SetFG( C SCR FRAME );
                                     /* draw screen frame */
     aRect( SCR_L + 1 , SCR_R + 1 , SCR_R - 1 , SCR_T - 1 , 0x40 );
    window_frame_drawa = 20;
legend_drawa = 20;
                                    /* window frame is gone */
                                     /* legend is gone */
    DMA alev drawn = NO;
                                     /* DMA elevations are gone */
add_layer ()
                                     /* add a layer to display */
    if ( stromp( vpame , ** ) == 0 )
                                                       /* if no active view, give up */
        SLOUTP ("choose a view first");
         stropy( layer2edd , "" );
        MENVALS ();
        return( -1 );
```

```
for ( i = 0 ; i < strlen( layer2add ) ; i++ ) /* assure name is uppercase */
if ( ( layer2add[i] >= 'a' ) && ( layer2add[i] <= 'E' ) )
layer2add[i] -= 040;</pre>
       if ( stromp( layer2add , "DMA_NLEV" ) == 0 ) /* special case */
            return(0);
       /* see if it exists */
       laytype = ' ';
      sprintf(fnem, "bs\\te\\te\\digit\\te",
gisdbase, mapset, location, layar2add);
if((fp = fopen(fnem, "r")) != NULL)
                                                                   /* digit file? */
      leytype = 'D';
if ( laytype = ' ')
                                                                   /* 20, cell file? */
           sprintf( famm, "%s\\%s\\%s\\call\\%s",
           gischese, mapset, location, layer2ndd);
if ( {fp = fopen( fnem , "r" ) } = monl()
laytype = 'C';
      if ( laytype -- ' ' )
                                                                   /* mo, doesn't exist */
           SLOOTS("Tadd_layer: no such layer %s", layer2edd);
stropy( layer2edd , "" );
           MENVALS () ;
           return ( -1 );
      folose( fp );
    if ( | del_layer_ia_progress )
      if ( num_dlayers >= MAX_DLAYERS ) /* add it to displayed-layer list */
          SLOUTS ("fadd_layer: screen list is full");
          stropy( layer2add , "" );
          MENVALS () ;
          return( -1 );
     stropy( dlams[sum_dlayers] , layer2sdd );
     aum_diayers++;
     build_legend();
                                                        /* add it to legend */
     if ( legend drawn )
         hide_legend();
         show_legend();
    SLOOP( "Drawing to" , layer2add );
    if ( laytype - 'D' )
                                                        /* draw it */
    d_digit();
else if ( laytype == 'C' )
         d cell();
    stropy( layerfadd , "" );
    MENVALS ();
del_layer ()
                           /* remove a layer from display */
    int i, idx;
    char save_ld;
   if ( stromp( layer2add , "DMA_ELEV" ) == 0 ) /* special case */
        DMA alev draws = 20;
   /* first remove it from displayed-layer list */
                                                                /* find it in list */
   for ( i = 0 ; i < num dlayers ; i++ )
if ( stromp( dlama[i] , layer2del ) == 0 )</pre>
```

```
14x = 1;
       1f ( 1dx = -1 )
            stoom ("layer to is not on display", layeridel);
stropy( layeridel , "" );
            MENVALS () ;
            setura ( -1 );
       if ( idz < num_dlayers - 1 )
                                                          /* if it wasn't last in list */
           for ( i = idx ; i < num_dlayers - 1 ; i++ )
    stropy( dlasm[idx] , dlasm[idx+1] );
stropy( dlasm[aum_dlayers-1] , "" );</pre>
                                                                     /* all move up */
                                                                      /* erase last */
      ava_dlayers--;
      MENVALE ();
                                                                      /* ... gratuitous */
        * Astually delete the layer from the graphics display.
       * For now, brute force: erase the graphics screen and
          repaint all layers but the deleted one.
      save_ld = legend_drawa; /* remember legend_drawa */
      erese_graphics_display();
      del_layer_in_progress = TES;
      for ( i = 0 ; i < num_dlayers ; i++ )
          stropy( layer2edd , dlasm[i] );
          if ( add_layer() = -1 )
               SLOUTS ("?del_layer: redraw error on %s" , layer2add );
               stropy( layer2del , "" );
               HEWVALS ();
              del_layer_in_progress = NO;
return( -1 );
     }
                                                /* recall legend drawn */
/* & redraw it if mecessary */
     legend_drawn = save_ld;
     if ( legend drawn )
         hide_legend();
show_legend();
     del_layer_in_progress = NO;
     stropy( layer2del , "" );
     MENVALE ():
store_screen ()
    /* if screen is blank, give up */
    if ( mm_dlayers -- 0 )
        SLOUTP("?store_screen: screen is empty -- mothing to save");
stropy( layersaveneme , "" );
         MENVALS ();
         return ( -1 );
    }
    /* if layer with this name already exists, give up \star/
    sprintf( fam, "%s\\%s\\%s\\digit\\%s",
   gisdbase, mapset, location, layerseveness
if ( (fp = fopen(fnem , "r" ) ) != NULL )
        SLOUTP("?store_screen: layer %s already exists", layersaveneme);
stropy( layersaveneme , "" );
        HERVALE ();
        return( -1 );
```

```
printf( fnam, "ts\\ts\\ts\\call\\ts",
    qisdhase, mapset, location, layersevename
if ( ( fp = fopen( fnam , "r" ) ) |= NOLL )
           SLOTT ("Tstore_screen: layer to already exists", layersevename);
stropy( layersevename , "" );
NEWWALS();
           moturn( -1 );
      laytype = 'C';
      screen2cell();
      add_to_layertst();
     stropy( layerseveneme , "" );
 show_dist ()
     SLOUTS ( "This feature is not yet available" );
 show_coords ()
     SLOUPS ( "This feature is not yet available" );
 edit_colors ()
     SLOUTS ( "This feature is not yet available" );
dlip_to_window ()
    SetADR( W1 , Wb , Wr , Wt );
clip_to_screen ()
    SetADR( SCR_L , SCR_R , SCR_R , SCR_R );
draw_des_alor ()
                                     /* special branch of add_layer() */
   int 1:
   if ( | del_layer_im_progress )
    if ( num_dlayers >= MAX_DLAYERS ) /* add it to displayed-layer list */
        SLOOPS ("Todd_Layer: screen list is full");
stropy( layer2add , "" );
        MENVALS ();
        return( -1 );
   stropy( dlass(sum_dlayers) , layer2edd );
   aum dlayers++;
   build_dms_elev_legend();
                                                       /* special legend too */
   IMA_elev_being_draws = TES;
   if ( stromp( vpasse , "Sells.G" ) == 0 )
        stropy( layer2add , "AJO_E.C" );
       SLOOP( "Drawing AJO E.C (DMA ELEV part 1 of 5)" );
        d_oali();
       stropy( layer2sdd , "AJO W.C" );
SLOOT( "Drawing AJO W.C (DAG ELEV part 2 of 5)" );
        d_col1();
       stropy( layer2add , "LUEEV.C" );
SLOUZ( "Drawing LUEEV.C (DEA ELEV part 3 of 5)" );
```

```
stropy( layer2edd , "MOGAL.C" );
           SLOUP( "Drawing MOGAL.C (MGA_MILEV part 4 of 5)" );
           d_cell();
           stropy( layer2add , "TOCSE.C" );
          SLOUZ( "Drawing TOCSE.C (DEC. ELEV part 8 of 5)" );
          d_oali();
      also if ( stromp( vpame , "Ajo.M" ) == 0 )
          stropy( layer2edd , "AJO_E.MA" );
SLOUT( "Drawing AJO_E.MA (DMG_ELSV part 1 of 1)" );
          d_coll();
      also if ( stromp( vpame , "Solls.N" ) == 0 )
          stropy( layer2add , "AJO_E.ms" );
          SLOUZ( "Drawing AJO E.MS (DMA ELEV part 1 of 4)" );
          d cell();
          stropy( layer2edd , "LUEEV.MS" );
          SLOUP( "Drawing LUXEV.MS (DMA_ELEV part 1 of 4)" );
          4_0011();
          stropy( layer2add , "MOGAL.MS" );
          SLOUZ( "Drawing MOGRAL.MS (INCL. MILEV part 1 of 4)" );
          d_coll();
          stropy(layer2add, "TOCSE.MS");
          SLOUZ ( "Drawing TOCSE.MS (DAGA ELEV part 1 of 4)" );
          4_0011();
      else if ( stromp( vpame , "Ajo.F" ) == 0 )
          stropy( layer2add , "AJC_E.FA" );
          MLOUT( "Drawing AJO E.FA (DMA MLEV part 1 of 1)" );
          d_coll();
     also if ( stromp( vpasse , "Salis.F" ) == 0 )
         stropy( layer2add , "MOGAL.Ps" );
         SLOUT ( "Drawing MOGAL.FS (DAM MLEV part 1 of 1)" );
         d_coll();
         SLOUTS ( "? no TMCA elevation data for this view" );
         DMA elev being drawn = NO;
DMA elev drawn = NO;
return( -1 );
     stropy( layer2edd , "" );
     MENVALS ();
    DMR_elev_being_drawn = NO;
DMR_elev_drawn = YES;
 dround ( aum )
                        /* double-precision-to-integer rounding routine */
    if ( num > 0.0 ) return (int) (num + 0.499);
else return (int) (num - 0.499);
/* and of grass.c */
ig.c -- device driver functions for Imagraph AGC hardware.
 *****************
        #define GRAPHICS DEBUGGING 1
finglade "stdio.h"
finclude "acrtc.h"
finclude "imagraph.h"
ist
       GDBA = 0, GDBP = 0;
                                        /* debug switches */
/* Imagraph hardware/software stuff */
```

d cell();

```
max_intensity;
                                 /* max color intensity for this hardware */
         *Palette;
  chez
                                 /* intensities of 1st 16 colors */
        *model;
  Cher
                                 /* handy pointer to IG model ident */
  /* fill petters mask */
 /* Stuff defined in Imagraph libraries */
 extern int
                FourBit;
                                 /* 4 ⇒ 4-bit, 0 ⇒ 8-bit */
 estern Distra *pDG;
 extern Dilstru Dilmodel[];
extern SCHOOLF BaseSC;
 extern SCHOOLF UpperSC;
extern SCHOOLF LowerSC;
 /* functions */
 void #Cala();
 Distra *Dispes();
 char *paialloc();
char *gotenv();
 void begin_Id_graphics ()
         DGstre *board:
        int m, ya;
 #1fdef GRAPHICS DESCRIPTION
         GDBA = 0; GDBP = 0;
        printf("\mannounce every hardware call? (1=yes,0=mo): ");
         sound("4d", 6000A);
         If ( GODA )
            printf("pauses after hardware call amountements? (1=yes,0=no): ");
            scenf("td", scens);
 #endif /* GRAPRICS_DESUGGING */
        model = getenv("INGMODEL");
        if ( | ( board = Diflopen (model) ) )
            \label{lem:cond}  \mbox{debugout($$^n$ Blackers act found $$n^*$);} 
            exit(0);
        tweaker();
                                       /* let's tweek hardware */
        = board->smampix;
        ym = board->ymempin;
        Opporsc.merz = 0;
                                       /* don't use upper sureen */
        UpperSC.alips = 0;
        OpposeC.many = 0;
        Opperso.olipy = 0;
        LowerSC.maxx = 0;
                                       /* dom't use lower sureen */
        LowerSC.alipx = 0;
        LowerSC.mary = 0;
        LowerSC.alipy = 0;
        BasesC.marx = m + 1;
                                       /* base sureen for graphics */
        BeselC.clipz = ma;
        Beseschery = ye + 1;
        BasesC.clipy = ya;
        SCalc(RESET_SCREEN, SReseSC); /* reset frame buffer pointer */
        SCale (BASE_SCREEN, &BaseSC);
       IG_bdw_init();
                                       /* initialise hardware */
       SetOrg( BaseSC.ol, BaseSC.o2); /* soord system on base sursen */
       max intensity = (FourEit) ? 15 : 255; /* max color intensity value */
       Palette = palallog();
                                     /* allocate palette image memory */
       load_default_pelette();
                                       /* load default pelette */
       CirSora ( &BeseSC , 0 ); ***/ /* clear graphics screen to MACE */
/***
       SetTG(7);
                                       /* default foreground is WEITE */
       Set3G( 0 );
                                       /* default background is BLACK */
       SetPata ( -0 );
                                       /* default petters is solid */
void end_IG_graphics ()
 (
       IG hdw_release();
                                       /* clean up hardware */
       free (Palette) ;
                                      /* clean up heep */
 }
```

A SAMPLE OF THE STREET, AND ADDRESS.

```
static IC hdw_init ()
                                  /* init hardware */
    (
Dikinit();
                                            /* initialise the herdware */
      SetADR(0,0,pDM->maxpix,pDM->maxpix);
imitCl(0,0,0,0);
      itzimode( 1 );
                                            /* get us into graphics for SGA */
      metura (0):
 static IC_hdw_release ()
                                            /* close hardware */
     itzimode( 0 );
                                           /* get us out of graphics for SGA */
     return (0) ;
 load_default_palette ()
     make_default_pelette();
     putlut ( Palette );
     return (0);
 static make_default_pelette ()
         ist mi, mi34, mi2;
         mi = (FourBit) 7 15 : 255:
         mi34 = (mi >> 2) * 3; /* 75% intensity */
         m12 = m1 >> 1;
                                       /* 504 intensity */
         pinitCI(Palette, 0. 0, 0, 0);
         pinitCI(Palette, 1, mi, 0, 0);
pinitCI(Palette, 2, mi, 0, mi);
         pinitCI(Palette, 3, 0, 0, mi);
         pinitCI(Palette, 4, 0, mi, mi);
pinitCI(Palette, 5, 0, mi, 0);
        pinitCI (Palette, 6, mi, mi, 0);
pinitCI (Palette, 7, mi, mi, mi);
pinitCI (Palette, 8, mi2, mi2, mi2);
         pinitCI(Palette, 9, mi34, 0, 0);
         pinitCI (Palette, 10, mi, 0, mi34);
         pinitCI(Felette, 11, 0, 0, mi34);
         pinitCI (Palette, 12, 0, mi, mi34);
        pinitCI(Palette, 13, 0, mi34, 0);
        pinitCI(Paletta, 14, mi34, 0, mi34);
pinitCI(Paletta, 15, mi34, mi34, mi34);
        return(0);
  3
/* end of ig.a */
leg.d -- ASAN GRASS legend and ostegory file heading.
 ************************
#define DB (str)
finalude. "grass.h"
finalude "gglobs.h"
Modine LBG_FRAME_SPACE
Modine LBG_FV_WIDTE
                                  ( charwidth / 4 )
                                  ( charwidth )
Modine LEG BY REIGHT
                                  ( chardepth - 2 )
#define LBG_PV2TXT_SPACE
                                 ( charwidth / 2 )
extern int
                chardepth, charwidth;
statio int
                leg_frume_top;
static ist
                leg_freme_bot;
static int
                log_frame_loft;
static ist
                log_frame_right;
                leg_frame_height;
static int
static int
                leg frame width;
```

-----

```
static char
                  fam [80];
  static FILE
                  *£p;
  static ist
                  sount;
  static ist
                  fline;
  static char
                  junk[100];
 build_legend ()
                       /* build legend data structure */
     int dlide, 1, 3;
     for ( dlidz = 0 ; dlidz < ava_dlayers ; dlidz++ )
          sprintf( fnam, "4s\\4s\\4s\\cats\\4s",
          gisdbase, mapset, location, dlass[dlidx]);
if ( (fp = fopen(fasm , "r" ) ) == NOLL )
              SLOUTS ("Thuild_leg: fopen %s failed", face);
             return( -1 );
          fline = 1;
          BLERR( "bed count or early BOF" );
             return( -1 );
          for (1 = 0; 1 < 2; 1++)
                                                          /* skip mest 2 lines */
             flime++;
             if ( fgets( junk , 100 , fp ) - NULL )
             {
    BLERK( "fgets error" );
                return( -1 );
         1
         for ( 1 = 0 ; 1 < count ; i++ )
                                                         /* the rest are cats */
             if ( num_llines >= MAX_CATS )
                                                         /* prevent overflow */
             {
    MADR( "legend data space full" );
                return( -1 );
             flize++;
                                                         /* store string */
             if ( fgets( leg[num_llines].str , 100 , fp ) == NULL )
             ( BLERR( "bad data or early BOF" );
               return ( -1 );
            /* replace <ret> is string with <aull> */
            for ( j = 0 ; j < strlen( leg[num_llines].str ) ; j++ )
if ( leg[num_llines].str[j] == '\n' )</pre>
                    leg[num_llines].str[j] = '\0';
            leg[num_llines].pv = next_ost_pv;
                                                    /* store pirvel */
            mest_cet_pv++;
        }
        folose( fp );
DB( printf("count=6d num_llines=6d:\n", dount, num_llines); )
DB( for ( i = 0 ; i < num_llines ; i++ ) )
DB( printf( "%s\n" , leg[i].str );
DB ( getch();
static BLERR (a, b, c, d, a, f, g)
    char magbuf[80];
    cher tmpbuf[80];
```

```
sprintf( tmpbuf , a , b , a , d , a , f , g );
sprintf( megbuf , "Tbuild_leg %s (%d): " , famm , fline );
stroat( megbuf , tmpbuf );
       SLOUTE ( magbel );
       falose (fp);
   show_legend ()
                             /* draw legend on graphics screen */
       int 1, 5, wid, y;
       if ( legend_drawn )
                                              /* if legend is visible, erase it */
           hide_legued();
      leg_freme_top = W_b - 4;
leg_freme_left = 8;
       wid = 0;
                                               /* find widest legend line string */
       for ( i = 0 ; i < num_llines ; i++ )
           12 ( ( ) = strlen( leg[i].str ) ) > wid )
               wid = j;
      log_frame_height = ( xmm_llines * chardepth ) + ( LBG_FRAME_space * 2 );
      log_frame_width = LBG_FV_WIDTE + LBG_FV2TXT_SPACE
                          + ( wid * charwidth ) + ( INC_FRAME_SPACE * 2 );
      leg_frume_right = leg_frume_left + leg_frume_width;
      if ( ( leg_frame_bot = leg_frame_top - leg_frame_height ) < 0 )
          SLOUZE ("?legend frame below surem hottom");
          leg_frame_bot = 1;
      SetIG( C_LEG_IRAME );
                                                                /* draw frame */
     alect( leg_frame_left , leg_frame_bot , leg_frame_right , leg_frame_top , 0x40 );
     y = leg_frame_top - LBG FRAME_SPACE - chardepth;
                                                               /* draw legend lines */
      for ( i = 0 ; i < num_llines ; i++ )
          15 ( y < 0 )
              SLOUTS ("?legend text below screen bottom");
              legend_draws = YES;
              return ( -1 ):
         SetTG( leg[1].pv );
                                                               /* draw color block */
         affloot ( log_frome_left + LEG_FRAME_SPACE , y ,
           leg from left + LEG FRAME SPACE + LEG BY WIDTE , Y + LEG BY MEIGHT ,
         stropy( txt , leg[i].str ); /* draw string */
ixl = leg_frame_laft +LEG_FRAME_SPACE + LEG_FV_WIDTE + LEG_FV2TKT_SPACE;
         iy1 = y + ( cherdepth / 2 );
         setFG( C_LEG_TEXT );
         d_test ( SCHOOL COCKES );
        y - chardepth;
    }
    if ( DMA_alov_drawn ) show_DMA_alov_legend();
    legend_drawn = YES;
hide_legend ()
                         /* erase legend from graphics screen */
   if ( legend_drawn )
        SetFG( C_BES );
                                           /* IS THIS THE CORRECT CHO?? */
        afrect ( leg_frame_left , leg_frame_bot ,
                 leg_frame_right , leg_frame_top , 0x40 );
```

```
if ( DM elev_drawn )
         hide_DMA_elev_legend();
     legend_drawn = NO;
 build_dms_elev_legeed ()
                                 /* build special legend for DNA elevations */
                                        /* ADD ME LATER */
   }
 show_DMA_elev_legend ()
                                /* draw IMSA elevation legend */
                                        /* ADD ME LATER */
   }
 hide_DMA_elev_legend ()
                                /* erase TMA elevation legend */
   (
                                        /* ADD ME LATER */
   )
 /* end of leg.c */
 * lt.s. -- ASAN GRASS map layer list/description text file menage
          NOTE: Home "layers.txt" is wired into this code.
  finolude "grass.h"
finolude "gglobs.h"
 static char
                fam [80];
 static FILE
               *fp;
nambuf[12];
 static char
add_to_layertzt ()
                               /* add new layer name to the file */
    int 1;
    if ( ( fp = fopen( "layers.txt" , "r" ) ) = NULL ) /* first find it */
        SLOUTS ("fadd21: fopen(r] layers.txt failed");
       return ( -1 );
    folose(fp); '
    if ( ( fp = fopen( "layers.txt" , "a" ) ) \Longrightarrow NVLL ) /* now really open it */
       SLOOTS ("Tadd21: fopen[a] layers.txt failed");
       return ( -1 );
   /* assure layer name is uppercase */
   for (i = 0; i < strlen(layersaveneme); i++)
if ((layersaveneme[i] >= 's') 46 (layersaveneme[i] <= 's'))
nembuf[i] = layersaveneme[i] - 040;</pre>
       alse
           numbef[i] = layersavename[i];
   mambef[++1] = '\0';
   fprintf( fp , "%s\n" , membuf );
   falose(fp);
rem_from_layertxt ()
                              /* remove a layer name from the file */
```

```
/* and of 1t.a */
   /* ADD BETTER FIRED CHECKING. SAVE WIN & VP, CHECK AGAINST NEW CHES */
   ww.d -- read winddow & viewport files, set up display parameters
                  FILE POPMETS ARE EARDWINED INTO THIS CODE,
                   AND ORDER OF ITMES IN FILES IS CRUCIAL.
    finaledo "gress.h"
  finalude "gglobe.h"
  finciade cfosti.b>
  fincinde 
  fincinde (sys/stat.)
  finaleds cio.b>
  static char
                 fnem [80] :
  static cher
                 junk [20] ;
  static FILE
                 *£p;
  set_display_params ()
                                         /* set up display permeters */
     double h_temp, v_temp;
     /* read the window file */
     sprintf( fam, "ts\\ts", gisdbase, winneme);
     if ( (fp = fopen(fam , "x" ) ) == MULL )
         SLOUTS ("Tww: fopen window to failed", winnesse);
         return ( -1 );
    fecanf( fp , "4s 4d" , junk , 4W_1 );
fecanf( fp , "4s 4d" , junk , 4W_r );
fecanf( fp , "4s 4d" , junk , 4W_b );
fecanf( fp , "4s 4d" , junk , 4W_t );
    falose ( fp);
    /* Find number of rows & columns in screen window */
    W_REONS = W_t - W_b + 1;
    W_mools = W_r - W_1 + 1;
DB(SLOUZE)("TW: W_totd W_botd W_lotd W_rotd W_nrowsetd W_ncolsetd", W_t, W_b, W_l, W_r, W_nrows, W_ncols);)
    if ( | window_frame_drawa )
       SetFG( C WIN FRAME );
       aRect(W1-1,Wb-1,Wr+1,Wt+1,Ox40);
window_frame_drawa = YMS;
   /* read the viewport file */
   sprintf( famm, "%s\\%s\\%s\\vwport\\%s",
     gischese, mepset, location, vpmsme);
   if ( ( fp = fopen( fnom , "r" ) ) - NOLL )
      SLOUZE ("Tww: fopen vmport to failed", vpasse);
      return( -1 );
   fromf(fp , "to td"
                           , junk ,
                                          &V_proj );
   fsoanf( fp , "ts td"
                           , junk ,
                                          AV_Ecos );
   frommf( fp , "to tif"
                           , junk ,
  &V_north );
```